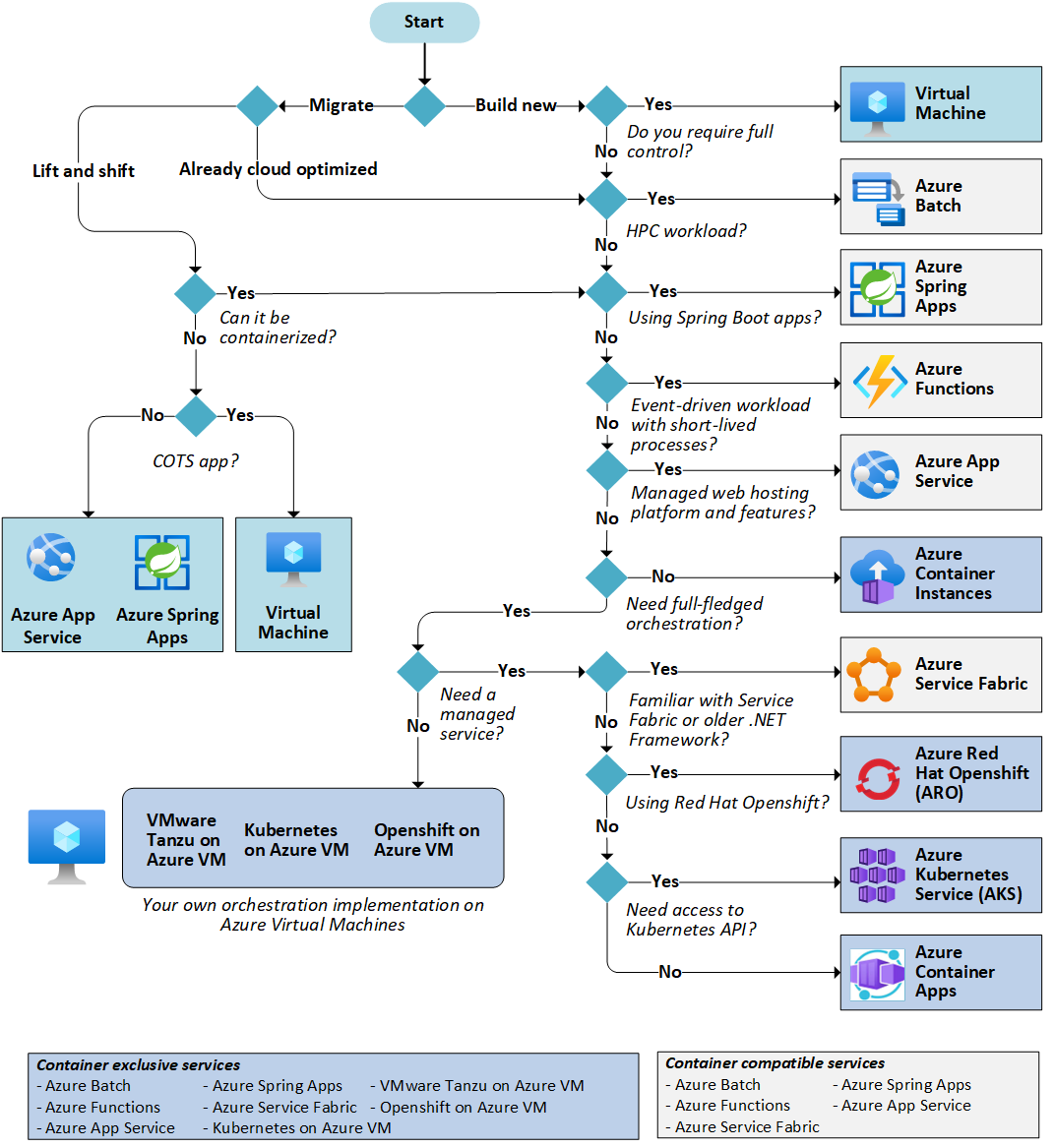
# **Sizes for virtual machines in Azure**

| **Type** | **Sizes** | **Description** |
| --- | --- | --- |
| [General purpose](https://learn.microsoft.com/en-us/azure/virtual-machines/sizes-general) | B, Dsv3, Dv3, Dasv4, Dav4, DSv2, Dv2, Av2, DC, DCv2, Dpdsv5, Dpldsv5, 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](https://learn.microsoft.com/en-us/azure/virtual-machines/sizes-compute) | F, Fs, Fsv2, FX | High CPU-to-memory ratio. Good for medium traffic web servers, network appliances, batch processes, and application servers. |
| [Memory optimized](https://learn.microsoft.com/en-us/azure/virtual-machines/sizes-memory) | Esv3, Ev3, Easv4, Eav4, Epdsv5, Epsv5, Ev4, Esv4, Edv4, Edsv4, Ev5, Esv5, Edv5, Edsv5, Easv5, Eadsv5, Mv2, M, DSv2, Dv2 | High memory-to-CPU ratio. Great for relational database servers, medium to large caches, and in-memory analytics. |
| [Storage optimized](https://learn.microsoft.com/en-us/azure/virtual-machines/sizes-storage) | Lsv2, Lsv3, Lasv3 | High disk throughput and IO ideal for Big Data, SQL, NoSQL databases, data warehousing and large transactional databases. |
| [GPU](https://learn.microsoft.com/en-us/azure/virtual-machines/sizes-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](https://learn.microsoft.com/en-us/azure/virtual-machines/sizes-hpc) | HB, HBv2, HBv3, HC, H | Our fastest and most powerful CPU virtual machines with optional high-throughput network interfaces (RDMA). |

# **Choose an Azure compute service**

## **Choose a candidate service**

Use the following flowchart to select a candidate compute service.



## **Understand the basic features**

If you're not familiar with the Azure service selected in the previous step, read the overview documentation to understand the basics of the service.

* [Azure App Service](https://learn.microsoft.com/en-us/azure/app-service). A managed service for hosting web apps, mobile app back ends, RESTful APIs, or automated business processes.
* [Azure Spring Apps](https://learn.microsoft.com/en-us/azure/spring-apps). A managed service designed and optimized for hosting Spring Boot apps.
* [Azure Kubernetes Service](https://learn.microsoft.com/en-us/azure/aks/intro-kubernetes) (AKS). A managed Kubernetes service for running containerized applications.
* [Azure Batch](https://learn.microsoft.com/en-us/azure/batch/batch-technical-overview). A managed service for running large-scale parallel and high-performance computing (HPC) applications
* [Azure Container Instances](https://learn.microsoft.com/en-us/azure/container-instances/container-instances-overview). The fastest and simplest way to run a container in Azure, without having to provision any virtual machines and without having to adopt a higher-level service.
* [Azure Functions](https://learn.microsoft.com/en-us/azure/azure-functions/functions-overview). A managed FaaS service.
* [Azure Service Fabric](https://learn.microsoft.com/en-us/azure/service-fabric/service-fabric-overview). A distributed systems platform that can run in many environments, including Azure or on premises.
* [Azure Virtual machines](https://learn.microsoft.com/en-us/azure/virtual-machines). Deploy and manage VMs inside an Azure virtual network.
* [Azure Container Apps](https://learn.microsoft.com/en-us/azure/container-apps). A managed service built on Kubernetes, which simplifies the deployment of containerized applications in a serverless environment.
* [Azure Red Hat OpenShift](https://learn.microsoft.com/en-us/azure/openshift). A fully managed OpenShift cluster for running containers in production with Kubernetes.

## **Understand the hosting models**

Cloud services, including Azure services, generally fall into three categories: IaaS, PaaS, or FaaS. (There's also SaaS, software-as-a-service, which is out of scope for this article.) It's useful to understand the differences.

Infrastructure-as-a-Service (IaaS) lets you provision individual VMs along with the associated networking and storage components. Then you deploy whatever software and applications you want onto those VMs. This model is the closest to a traditional on-premises environment, except that Microsoft manages the infrastructure. You still manage the individual VMs.

Platform-as-a-Service (PaaS) provides a managed hosting environment, where you can deploy your application without needing to manage VMs or networking resources. Azure App Service and Azure Container Apps are PaaS services.

Functions-as-a-Service (FaaS) goes even further in removing the need to worry about the hosting environment. In a FaaS model, you deploy your code and the service automatically runs it. Azure Functions is a FaaS service.

## **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 [SAP HANA](https://learn.microsoft.com/en-us/azure/virtual-machines/workloads/sap/hana-vm-operations-storage), 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 an OS Disk?** | No | No | Yes | Yes | Yes |

### **Ultra disk size**

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

—----------------------------------------------------------------------------------------------------------------------------

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

There is always a multitude of [design considerations](https://learn.microsoft.com/en-us/azure/architecture/reference-architectures/n-tier/linux-vm) 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

**Creating virtual machines using CLI :-**

az vm create --name argoid-prod3-host-0031 --resource-group argoid-prod3-resource-group-1 --location centralindia --image "OpenLogic:CentOS:7\_6-gen2:latest" --size Standard\_E4as\_v4 --authentication-type ssh --admin-username manjunath --ssh-key-values manju.pub --storage-sku Standard\_LRS --os-disk-size-gb 30 --vnet-name argoid-prod3-vpc --subnet argoid-prod3-vpc-subnet-1 --nsg argoid-prod3-nsg --private-ip-address 10.1.0.31 --public-ip-address ""

—----------------------------------------------------------------------------------------------------------------------------

## 

## **Create resource group**

az group create --name myResourceGroupVM --location eastus

## **Create virtual machine**

az vm create \

--resource-group myResourceGroupVM \

--name myVM \

--image UbuntuLTS \

--admin-username azureuser \

--generate-ssh-keys

## **Connect to VM**

ssh azureuser@52.174.34.95

## **Understand VM images**

az vm image list --output table

**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

### **Find available VM sizes**

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

### 

### **Create VM with specific size**

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 of size of a VM with** [**az vm show**](https://learn.microsoft.com/en-us/cli/azure/vm)**:**

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**](https://learn.microsoft.com/en-us/cli/azure/vm) **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**](https://learn.microsoft.com/en-us/cli/azure/vm) **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**](https://learn.microsoft.com/en-us/cli/azure/vm) **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. |

## **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**

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

—------------------------------------------------------------------------------------------------------------------

## **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](https://learn.microsoft.com/en-us/cli/azure/vm/disk) 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](https://learn.microsoft.com/en-us/azure/availability-zones/az-overview). 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

## **Attach an existing disk**

To attach an existing disk, find the disk ID and pass the ID to the [az vm disk attach](https://learn.microsoft.com/en-us/cli/azure/vm/disk) 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](https://learn.microsoft.com/en-us/azure/virtual-machines/linux/mac-create-ssh-keys). 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"

### **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.

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](https://xfs.wiki.kernel.org/) 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](https://linux.die.net/man/8/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

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

Next, open the */etc/fstab* file in a text editor as follows:

sudo nano /etc/fstab

UUID=33333333-3b3b-3c3c-3d3d-3e3e3e3e3e3e /datadrive xfs defaults,nofail 1 2

### **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

RHEL/CentOS

sudo yum install util-linux

sudo fstrim /datadrive

## **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 if 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

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**

az vm disk detach \

-g myResourceGroup \

--vm-name myVm \

-n myDataDisk

## **Expand an Azure Managed Disk**

### **Expand without downtime**

You now may be able to expand your managed disks without deallocating your VM.

This feature has the following limitations:

* Only available with particular VM SKUs.
* 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 or Premium SSD v2 disks.
* Not supported for shared disks.
* Install and use either:
  + The [latest Azure CLI](https://learn.microsoft.com/en-us/cli/azure/install-azure-cli)
  + The [latest Azure PowerShell module](https://learn.microsoft.com/en-us/powershell/azure/install-az-ps)
  + The [Azure portal](https://portal.azure.com/)
  + Or an Azure Resource Manager template with an API version that's 2021-04-01 or newer.

### **Expand Azure Managed Disk**

1. Operations on virtual hard disks can't be performed with the VM running. Deallocate your VM with [az vm deallocate](https://learn.microsoft.com/en-us/cli/azure/vm#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. View a list of managed disks in a resource group with [az disk list](https://learn.microsoft.com/en-us/cli/azure/disk#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](https://learn.microsoft.com/en-us/cli/azure/disk#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

3. Start your VM with [az vm start](https://learn.microsoft.com/en-us/cli/azure/vm#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

—------------------------------------------------------------------------------------------------------------------

# **Change the OS disk used by an Azure VM using the Azure CLI**

SWAP THE OS DISK

Use [az disk list](https://learn.microsoft.com/en-us/cli/azure/disk) 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](https://learn.microsoft.com/en-us/cli/azure/vm) to stop\deallocate the VM before swapping the disks.

az vm stop \

-n myVM \

-g myResourceGroup

Use [az vm update](https://learn.microsoft.com/en-us/cli/azure/vm#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](https://learn.microsoft.com/en-us/cli/azure/vm).

az vm start \

-n myVM \

-g myResourceGroup

Next steps

To create a copy of a disk, see [Snapshot a disk](https://learn.microsoft.com/en-us/azure/virtual-machines/linux/snapshot-copy-managed-disk).

# **Create a snapshot of a virtual hard disk**

Follow these steps to take a snapshot with the az snapshot create command and the --source-disk parameter. This example assumes that you have a VM called *myVM* in the *myResourceGroup* resource group. The code sample provided creates a snapshot in the same resource group and within the same region as your source VM.

1. Get the disk ID with [az vm show](https://learn.microsoft.com/en-us/cli/azure/vm#az-vm-show).

osDiskId=$(az vm show \

-g myResourceGroup \

-n myVM \

--query "storageProfile.osDisk.managedDisk.id" \

-o tsv)

2. Take a snapshot named *osDisk-backup* using [az snapshot create](https://learn.microsoft.com/en-us/cli/azure/snapshot#az-snapshot-create). In the example, the snapshot is of the OS disk. By default, the snapshot uses locally redundant standard storage. We recommend that you store your snapshots in standard storage instead of premium storage whatever the storage type of the parent disk or target disk. Premium snapshots incur additional cost.

az snapshot create \

-g myResourceGroup \

--source "$osDiskId" \

--name osDisk-backup

If you would like to store your snapshot in zone-resilient storage, you need to create it in a region that supports [availability zones](https://learn.microsoft.com/en-us/azure/availability-zones/az-overview) and include the optional --sku Standard\_ZRS parameter. A list of [availability zones](https://learn.microsoft.com/en-us/azure/availability-zones/az-region#azure-regions-with-availability-zones) can be found here.

3. Use [az snapshot list](https://learn.microsoft.com/en-us/cli/azure/snapshot#az-snapshot-list) to verify that your snapshot exists.

az snapshot list \

-g myResourceGroup \

-o table

# **Create a virtual machine from a snapshot with CLI**

## **Sample script**

# <FullScript>

#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

# </FullScript>

## **Clean up deployment**

Run the following command to remove the resource group, VM, and all related resources.

az group delete --name myResourceGroup

—------------------------------------------------------------------------------------------------------------------

# **Enable or disable a firewall rule on an Azure VM Guest OS**

This article provides a reference for troubleshooting a situation in which you suspect that the guest operating system firewall is filtering partial traffic on a virtual machine (VM). This could be useful for the following reasons:

* If a change was deliberately made to the firewall that caused RDP connections to fail, using the Custom Script Extension feature can resolve the issue.
* Disabling all firewall profiles is a more foolproof way of troubleshooting than setting the RDP-specific firewall rule.

## **Solution**

How you configure the firewall rules depends on the level of access to the VM that’s required. The following examples use RDP rules. However, the same methods can be applied to any other kind of traffic by pointing to the correct registry key.

### **Online troubleshooting**

#### **Mitigation 1: Custom Script Extension**

1. Create your script by using the following template.

To enable a rule:

netsh advfirewall firewall set rule dir=in name="Remote Desktop - User Mode (TCP-In)" new enable=yes

To disable a rule:

netsh advfirewall firewall set rule dir=in name="Remote Desktop - User Mode (TCP-In)" new enable=no

<https://learn.microsoft.com/en-us/troubleshoot/azure/virtual-machines/enable-disable-firewall-rule-guest-os>