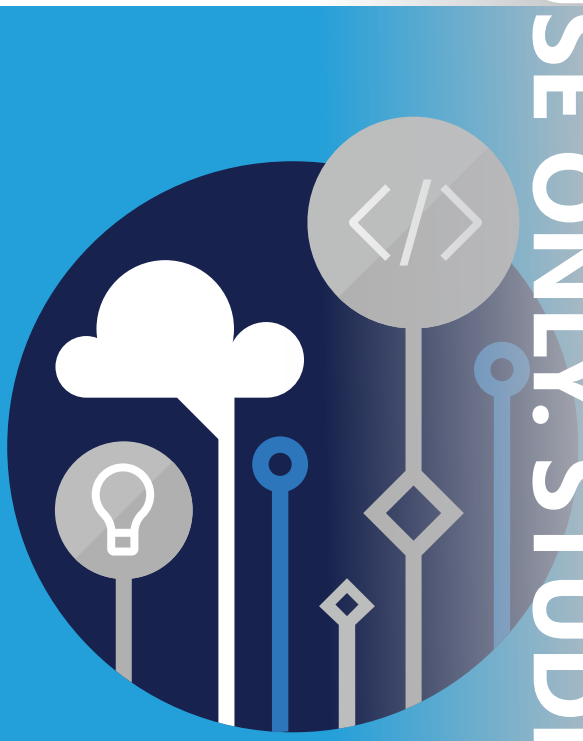
Microsoft

Microsoft Official Course



AZ-300T02

Implementing Workloads and Security

AZ-300T02
Implementing Workloads and Security

Contents

Module 0 Start Here

Welcome to Implementing Workloads and Security

Welcome to Understanding Implementing Workloads and Security

Course Overview: Implementing Workloads and Security

Welcome to *Implementing Workloads and Security* (AZ-300t2).This course is part of a series of six courses to help students prepare for Microsoft's Azure Solutions Architect technical certification exam AZ-300: Microsoft Azure Architect Technologies. These courses are designed for IT professionals and developers with experience and knowledge across various aspects of IT operations, including networking, virtualization, identity, security, business continuity, disaster recovery, data management, budgeting, and governance.

This course teaches IT professionals how to discover, assess, plan and implement a migration of on-premises resources and infrastructure to Azure. Students will learn how to use Azure Migrate to perform the discovery and assessment phase that is critical to a successful migration. They will also learn how to use Azure Site Recovery for performing an actual migration of workloads to Azure. This course focuses on using ASR on a Hyper-V infrastructure to prepare and complete the migration process.

Also, you will learn how to deploy serverless computing features like Azure Functions, Event Grid, and Service Bus. You will see how Azure multi-factor authentication facilitates safeguard access to data and applications, thus helping to meet customer demands for a simple sign-in process. Also, see how to use Azure Active Directory Privileged Identity Management to manage, control, and monitor access to Azure resources within an organization.

Additionally, learn how to manage and maintain infrastructure for core web apps and services that developers build and deploy. Discover how the Azure App Service is used as a Platform as a Service (PaaS) offering for deploying cloud apps for web and mobile environments.

Lastly, you will see how to implement advanced networking features, such as Application Gateway, and how to configure load balancing. See how to integrate on-premises networks with Azure virtual networks and use Network Watcher to monitor and troubleshoot issues.

The outline for this course is as follows:

Module 1 - Evaluating and Performing Server Migration to Azure

This module covers migrating workloads to a new environment, whether it be another datacenter, or to a public cloud, and setting clear goals for the migration. Goals include both technology-focused and business-focused goals for migrations, and how that benefits an organization's business. Activities include components of the Azure migration process: creating a project, creating a collector, assessing readiness, and estimating costs. Additionally, you will receive an overview of Azure Site Recovery (ASR) that includes end-to-end scenarios.

Module 2 - Implementing and Managing Application Services

This module includes the following topics:

- Deploying Web Apps
- Managing Web Apps
- App Service Security
- Serverless Computing Concepts
- Managing Event Grid
- Managing Service Bus
- Managing Logic App

Module 3 - Implementing Advanced Virtual Networking

This module includes the following topics:

- Azure Load Balancer
- Azure Application Gateway
- Site-to-Site VPN Connections

As well as an overview of ExpressRoute which allows companies to extend on-premises networks into the Microsoft cloud over a dedicated private connection facilitated by a connectivity provider.

Module 4 - Determining Azure Workload Requirements

A fictitious case study of Contoso, a US-based financial company based in Boston, where there are three additional local branches across the United States. The main datacenter is connected to the internet with a fiber metro Ethernet connection (500 Mbps). Each branch is connected locally to the internet using business class connections, with IPSec VPN tunnels back to the main datacenter. This allows the entire network to be permanently connected, and optimizes internet connectivity.

The module contains the following topics:

- Overview of Customer Case Study
- Step-by-Step: Determining Azure Workload Requirements
- Checklist of Assessment Goals

What You'll Learn

- Evaluating and Performing Server Migration to Azure
- Implementing and Managing Application Services
- Implementing Advanced Virtual Networking
- Securing Identities using Azure

Prerequisites

Successful Cloud Solutions Architects begin this role with practical experience with operating systems, virtualization, cloud infrastructure, storage structures, billing, and networking.

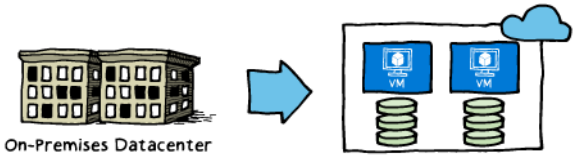
Module 1 Module Evaluating and Performing Server Migration to Azure

Migrate to Azure

Video: Migration

Migration Goals

When migrating any workload to a new environment, whether it be another datacenter, or to a public cloud, you should have a clear set of goals for migration in mind. Note that there are both technology-focused and business-focused goals that motivate potential migrations, but any such effort should result in direct benefits to the organization's business.



Some example goals that cause workload owners to consider migrations include:

- **Addressing the hardware obsolescence cycle**. Networking, storage, and compute hardware typically has a 3-5 years lifespan. After that time, the hardware becomes increasingly expensive to maintain.

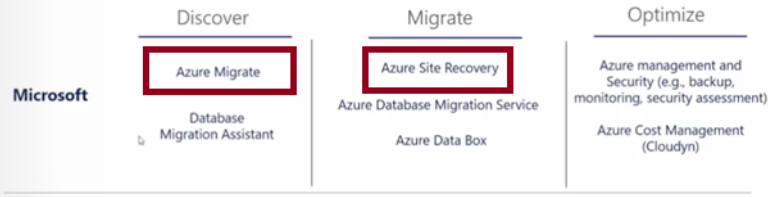
When new hardware is ordered, and software migrated the cycle starts again. Many organizations want a way out of this expensive planned obsolescence cycle.

- **Moving away from the 'pre-purchase capacity' model**. When purchasing hardware, organizations must pre-purchase enough capacity to grow into over a 3 to 5-year period. Organizations desire the ability to pay only for the capacity required at that moment, and to be able to scale workloads up, down, in, and out as demand dictates.
- **Lack of IT agility**. A perceived slowness in IT responding to business needs, can translate into missed opportunities. Organizations want to have IT respond quickly with robust, modern solutions when a business opportunity presents itself.
- **Desire to re-focus on core competencies**. Organizations whose core purpose is not related to managing complex datacenter deployments, may eventually want to shed competing interests and focus on improving their core business.
- **Expense of maintaining a global presence**. Organizations that have customers all over the world want to serve that distributed user base well. But maintaining datacenter deployments in many, geographically dispersed locations, is complex and expensive.
- **Enable disaster-recovery scenarios**. Business continuity and disaster recovery are critical concerns that keep business leaders up at night. But enabling these scenarios has typically been prohibitively expensive and extremely complex.

✓ What are your goals and reasons for migrating to the cloud?

Migration Phases

When planning for migration of workloads to Azure, consider three phases: Discover, Migrate, and Optimize.



Discover. In the Discover phase you work to get better visibility of applications, workloads, and data in your environment, and assess the optimal resource level to run them in Microsoft Azure. Use this information to help decide which workloads to move. Azure Migrate is the primary tool for this, and includes:

- Automated server, app, and database discovery.
- Intelligent workload right-sizing and costing for maximum ROI.
- Workload configuration analyses and recommendations.

Migrate. In the Migrate phase you move selected workloads to Azure. There are a variety of sources including physical servers and virtualized workloads hosted in Hyper-V or VMware environments. Azure Site Recovery is the primary tool in this area and includes:

- Lifting and shifting of servers, apps, databases, and data.
- Containerization of existing applications and infrastructure.
- Modernization options for apps and databases.

Optimize. In the Optimize phase you fine tune your Azure-based workloads and maximize your ROI. There are many Microsoft partners to help you with backup, monitoring, security assessments, and cost management.

✓ This course primarily emphasizes the first and second phases of a migration, focusing on Azure Migrate and Azure Site Recovery.

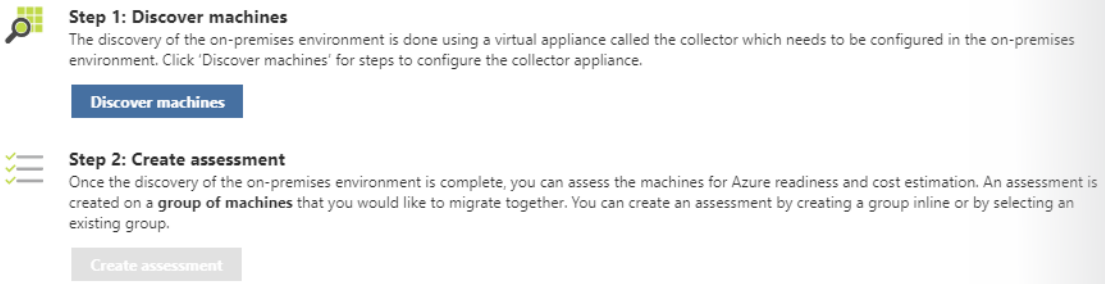
For more information, you can see:

Azure migration partners - **https://azure.microsoft.com/en-us/migration/partners/**

Azure Migrate Service

The Azure Migrate service assesses on-premises workloads for migration to Azure. The service examines the migration suitability of on-premises machines, performance-based sizing, and provides cost estimations for running your on-premises machines in Azure. If you're contemplating lift-and-shift migrations, or are in the early assessment stages of migration, this service is for you. After the assessment, you can use services such as **Azure Site Recovery**[1] and **Azure Database Migration Service**[2], to migrate the machines to Azure.

Azure Migrate follows two simple steps: Discover machines and Create assessment.



These steps will help you to:

- **Assess Azure readiness**. Assess whether your on-premises machines are suitable for running in Azure.
- **Get size recommendations**. Get size recommendations for Azure VMs based on the performance history of on-premises VMs.
- **Estimate monthly costs**. Get estimated costs for running on-premises machines in Azure.
- **Migrate with high confidence**. Visualize dependencies of on-premises machines to create groups of machines that you will assess and migrate together.

✓ Currently, you can only assess on-premises VMware virtual machines (VMs) for migration to Azure VMs. The VMware VMs must be managed by vCenter Server (version 5.5, 6.0, or 6.5).

For more information, you can see:

About Azure Migrate - **https://docs.microsoft.com/en-us/azure/migrate/migrate-overview**

Azure Migrate pricing - **https://azure.microsoft.com/en-in/pricing/details/azure-migrate/**

1 https://docs.microsoft.com/azure/site-recovery/site-recovery-overview
2 https://docs.microsoft.com/azure/dms/dms-overview

Video - Azure Migrate

You might be interested in downloading the **Migrate Your Virtual Machines to Microsoft Azure Proof of Concept guide**[3]. In addition to covering how Azure Migrate works it has information on partner Cloudamize and Movere solutions.

Video - Migrating Applications to the Cloud

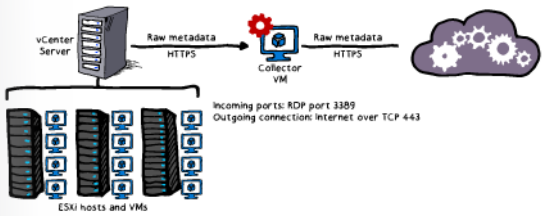
You might be interested in downloading the **Migrate Your Virtual Machines to Microsoft Azure Proof of Concept guide**[4]. In addition to covering how Azure Migrate works it has information on partner Cloudamize and Movere solutions.

Using Azure Migrate: A Look ahead

We've talked about the concepts and use cases for Azure Migrate. This topic serves to introduce the next lesson, where we'll look at using Azure Migrate to discover and assess on-premises workloads for migration to Azure.

Architecture

The basic architecture of the Azure Migration service is shown in the following diagram. Azure Migrate service works to discover information about ESXi hosts and VMs in a VMWare vCenter server. An assessment is created as an outcome of the discovery process.



Process

3 https://download.microsoft.com/download/F/4/1/F41EF1A8-98FB-4A6F-B535-95B675911ACC/PoC-guide-VM-Migration-to-Azure-final.pdf
4 https://download.microsoft.com/download/F/4/1/F41EF1A8-98FB-4A6F-B535-95B675911ACC/PoC-guide-VM-Migration-to-Azure-final.pdf

Here are the basic steps.

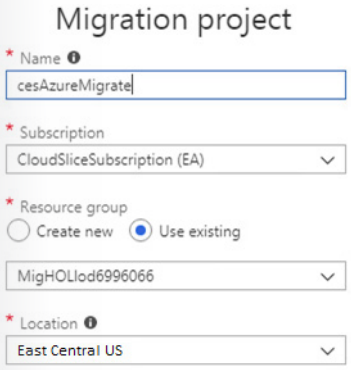
- **Create a project**. In Azure, create an Azure Migrate project.
- **Discover the machines**. Azure Migrate uses an on-premises VM called the collector appliance, to discover information about your on-premises machines. To create the appliance, you download a setup file in Open Virtualization Appliance (.ova) format and import it as a VM on your on-premises vCenter Server. You connect to the VM using console connection in vCenter Server, and then run the collector application in the VM to initiate discovery.
- **Collect the information**. The collector collects VM metadata using VMware PowerCLI cmdlets. Discovery is agentless and doesn't install anything on VMware hosts or VMs. The collected metadata includes VM information (cores, memory, disks, disk sizes, and network adapters). It also collects performance data for VMs, including CPU and memory usage, disk IOPS, disk throughput (MBps), and network output (MBps).
- **Assess the project**. The metadata is pushed to the Azure Migrate project. You can view it in the Azure portal. For the purposes of assessment, you can gather the discovered VMs into groups. For example, you might group VMs that run the same application. For more precise grouping, you can use dependency visualization to view dependencies of a specific machine, or for all machines in a group and refine the group. Once your group is formed, you create an assessment for the group. After the assessment finishes, you can view it in the portal, or download it in Excel format.

✓ The next lesson will go into each step in detail. At the end of that lesson, you will have an opportunity to practice the steps.

Azure Migrate Process

Creating a Project

You can create the Azure Migrate project from the portal. It is as simple as filling out Name, Subscription, Resource Group, and Location. The Azure Migrate project holds the metadata of your on-premises machines and enables you to assess migration suitability.



You can discover up to 1500 VMs in a single discovery and up to 1500 VMs in a single project. Additionally, you can assess up to 1500 VMs in a single assessment. If you want to discover a larger environment you can split the discovery and create multiple projects, **learn more**[5]. Azure Migrate supports up to 20 projects per subscription.

✓ Currently, you can only create an Azure Migrate project in West Central US or East US region. However, this does not impact your ability to plan your migration for a different target Azure location. The location of the migration project is used only to store the metadata discovered from the on-premises environment.

For more information, you can see:

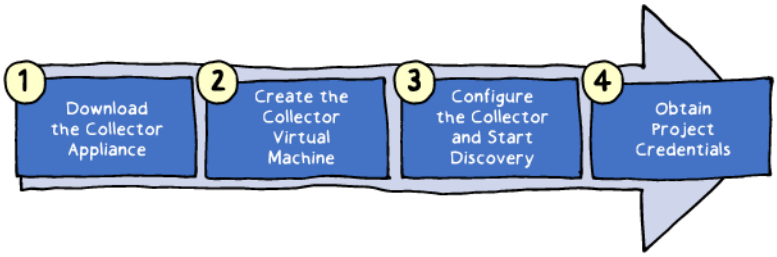
Create a project - **https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware#create-a-project**[6]

Current limitations - **https://docs.microsoft.com/en-us/azure/migrate/migrate-overview#current-limitations**[7]

Creating a Collector

The discovery of the on-premises environment is done using a virtual appliance called the Collector. The Collector is configured in the on-premise environment. There are four basic steps.

5 https://docs.microsoft.com/en-us/azure/migrate/how-to-scale-assessment
6 https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware
7 https://docs.microsoft.com/en-us/azure/migrate/migrate-overview



1. **Download the Collector Appliance**. The Collector appliance is a single file in Open Virtualization Appliance (.ova) format that you download from the Azure Migrate project on the Discovered Machines blade.
2. **Create the Collector Virtual Machine**. After the download is complete you can use the vCenter Server to import the file and create the Collector machine.
3. **Configure the Collector and Start Discovery**. Once the machine is created you can connect to collector and run the appliance using the shortcut on the desktop.
4. **Obtain Project Credentials**. The Collector will need the Azure Migrate project ID and key. This enables the Collector to send the discovered metadata to the appropriate Azure Migrate project. Once the Collector is running you can view the metadata being collected from the VMs. Typically, for 100 VMs, the collector takes around an hour for discovery to finish.

✓ The discover done by the compliance is a one-time discovery. This means that changes in the on-premises environment are not reflected once the discovery is complete. To re-discover new machines, connect to the Collector, and run it again.

For more information, you can see:

Download the collector appliance - **https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware#download-the-collector-appliance**

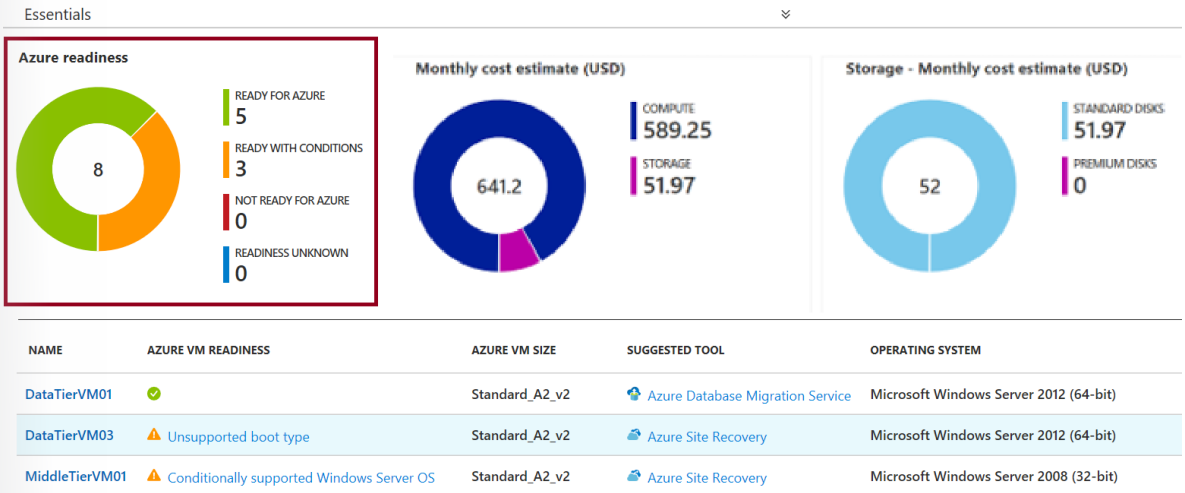
Create the Collector VM - **https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware#create-the-collector-vm**[8]

Run the collector to discover VMs - **https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware#run-the-collector-to-discover-vms**[9]

Assessing Readiness

Once your virtual machines have been discovered you can begin assessment. One of the assessment areas is Azure readiness.

8 https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware
9 https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware



The Azure readiness view in the assessment shows the readiness status of each VM. Depending on the properties of the VM, each VM can be marked as:

- **Ready for Azure (green)**. For VMs that are ready, Azure Migrate recommends a VM size in Azure. The next topic covers how the recommended size is determined.
- **Ready with conditions (Orange) and Not ready for Azure (Red)**. For these VMs, Azure Migrate explains the readiness issues and provides remediation steps. Several things are considered in making this determination: boot type, cores, memory, storage disks, networking components, and operating system. For example, a machine with an older version of Windows Server OS might be not ready for Azure. Read more at the reference link.
- **Readiness unknown (Blue)**. The VMs for which Azure Migrate cannot identify Azure readiness (due to data unavailability) are marked as readiness unknown. For example, a VM that was offline.

✓ Use the reference link for help troubleshooting readiness issues. For example, what to do if the disk count or disk size exceeds the limits.

For more information, you can see:

Azure suitability analysis - **https://docs.microsoft.com/en-us/azure/migrate/concepts-assessment-calculation#azure-suitability-analysis**[10]

Assessing VM Sizing

After a machine is marked ready for Azure, Azure Migrate sizes the VM and its disks for Azure. By default, Azure Migrate uses performance-based sizing. This is the best method when you may have over-allocated the on-premises VM, the utilization is low, and you would like to right-size the VMs in Azure to save cost.

For performance-based sizing, Azure Migrate starts with the disks attached to the VM, followed by network adapters, and then maps an Azure VM based on the compute requirements of the on-premises VM.

- **Storage**. Azure Migrate tries to map every disk attached to the machine to a disk in Azure.
- **Network**. Azure Migrate tries to find an Azure VM that can support the number of network adapters attached to the on-premises machine and the performance required by these network adapters.

10 https://docs.microsoft.com/en-us/azure/migrate/concepts-assessment-calculation

- **Compute**. After storage and network requirements are calculated, Azure Migrate considers CPU and memory requirements to find a suitable VM size in Azure.

Azure Migrate collects performance history of on-premises VMs from the vCenter Server. To ensure accurate right-sizing, ensure that the statistics setting in vCenter Server is set to level 3 and wait for at least a day before kicking off discovery of the on-premises VMs. If the statistics setting in vCenter Server is below level 3, performance data for disk and network is not collected.

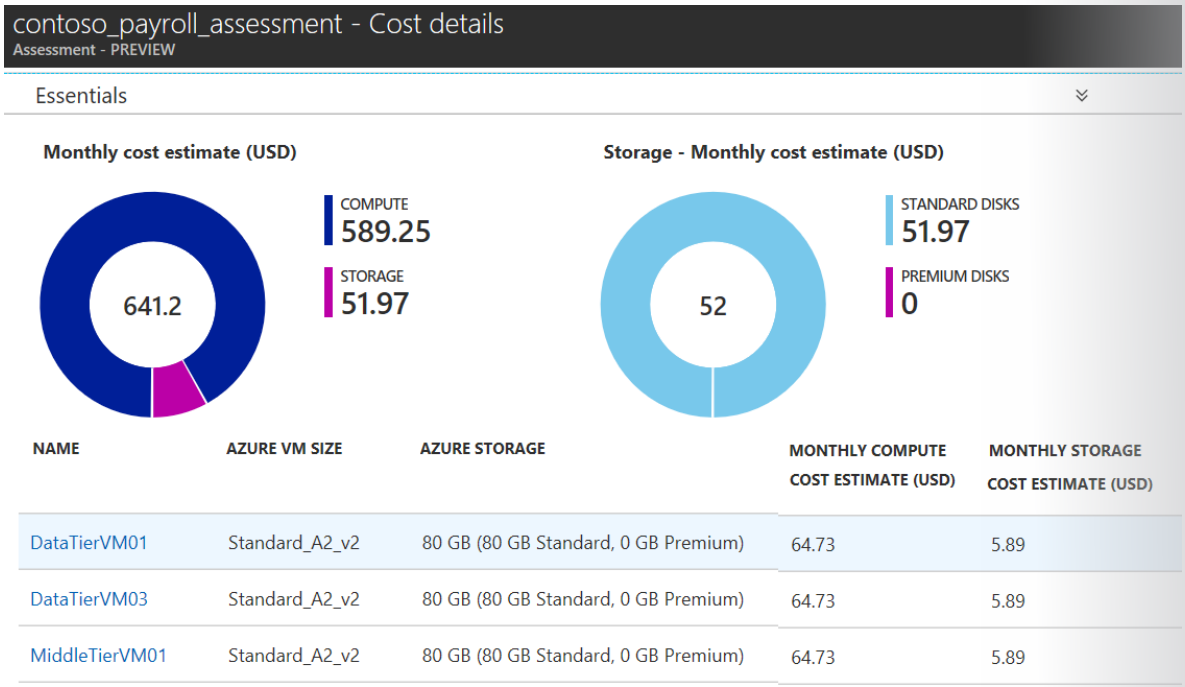
✓ The alternative to performance based sizing is as on-premises sizing. Azure Migrate allocates a VM SKU in Azure based on the size allocated on-premises. Similarly, for disk sizing, it looks at the Storage type and recommends the disk type accordingly. The default storage type is Premium disks.

For more information, you can see:

Sizing - **https://docs.microsoft.com/en-us/azure/migrate/concepts-assessment-calculation#sizing**[11]

Estimating Cost

The Cost assessment view shows the total compute and storage cost of running the VMs in Azure along with the details for each machine. Cost estimates are calculated considering the size recommendations done by Azure Migrate for a machine, its disks, and the assessment properties. Estimated monthly costs for compute and storage are aggregated for all VMs in the group.



Compute cost. Using the recommended Azure VM size, Azure Migrate uses the Billing API to calculate the monthly cost for the VM. The calculation takes the operating system, software assurance, reserved instances, VM uptime, location, and currency settings into account. It aggregates the cost across all machines, to calculate the total monthly compute cost.

Storage cost. The monthly storage cost for a machine is calculated by aggregating the monthly cost of all disks attached to the machine. Azure Migrate calculates the total monthly storage costs by aggregating the storage costs of all machines. Currently, the calculation doesn't take offers specified in the assessment settings into account.

11 https://docs.microsoft.com/en-us/azure/migrate/concepts-assessment-calculation

✓ The cost estimation provided by Azure Migrate is for running the on-premises VMs as Azure Infrastructure as a service (IaaS) VMs. Azure Migrate does not consider any Platform as a service (PaaS) or Software as a service (SaaS) costs.

For more information, you can see:

Monthly Cost Estimate - **https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware#monthly-cost-estimate**[12]

Customize the Assessment

Assessments can be customized based on your needs by changing the properties of the assessment. For example, changing from performance-based to as on-premises sizing estimates.

Setting	Details	Default
Target location	The Azure location to which you want to migrate. Azure Migrate currently supports 30 regions.	West US 2 is the default location.
Pricing tier	You can specify the **pricing tier (Basic/Standard)** *(https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes-general)* for the target Azure VMs.	By default the **Standard** *(https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes-general)* tier is used.
Storage type	You can specify the type of disks you want to allocate in Azure.	The default value is Premium managed disks.
Comfort factor	Azure Migrate considers a buffer (comfort factor) during assessment. This buffer is applied on top of machine utilization data for VMs.	Default setting is 1.3x.

✓ There are other options you might be interested in like currency, discounts, VM uptime, and performance history duration. Read more at the reference link.

For more information, you can see:

What's in an assessment? - **https://docs.microsoft.com/en-us/azure/migrate/migrate-overview#whats-in-an-assessment**[13]

Products available by region - **https://azure.microsoft.com/en-us/global-infrastructure/services/**

Troubleshooting Azure Migrate

Most issues with Azure Migrate involve collecting information. Here are a few of the most common errors.

- **Migration project creation failed**. This issue can happen when users do not have access to the Azure Active Directory (Azure AD) tenant of the organization. Users need to go to the email and accept the invitation to be added to the tenant.

12 https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware
13 https://docs.microsoft.com/en-us/azure/migrate/migrate-overview

- **No performance data**. This can occur if the statistics setting level on the vCenter server is set to less than three. At level three or higher, vCenter stores VM performance history for compute, storage, and network. For less than level three, vCenter doesn't store storage and network data, but CPU and memory data only.
- **Collector is not able to connect to the internet**. This can happen when the machine you are using is behind a proxy. Make sure you provide the authorization credentials if the proxy needs one.
- **Date and time synchronization error**. The server clock might be out-of-synchronization with the current time by more than five minutes. Change the clock time on the collector VM to match the current time.
- **Error UnableToConnectToServer**. There was no endpoint listening at https://Servername.com:9443/sdk that could accept the message. Check if you are running the latest version of the collector appliance, if not, upgrade the appliance.

✓ If you have an error code, you can look it up at the reference link. The link also has information on collecting logs.

For more information you can see:

Troubleshoot common errors - **https://docs.microsoft.com/en-us/azure/migrate/troubleshooting-general#troubleshoot-common-errors**[14]

Azure Migrate forum - **https://social.msdn.microsoft.com/Forums/en-US/home?forum=AzureMigrate**

Demonstration: Azure Migrate

Practice: Discover and Assess

Take a few minutes and try **Discover and assess on-premises VMware VMs for migration to Azure**[15] tutorial. In this tutorial, you learn how to:

- Create an account that Azure Migrate uses to discover on-premises VMs
- Create an Azure Migrate project.
- Set up an on-premises collector virtual machine (VM), to discover on-premises VMware VMs for assessment.
- Group VMs and create an assessment.

14 https://docs.microsoft.com/en-us/azure/migrate/troubleshooting-general
15 https://docs.microsoft.com/en-us/azure/migrate/tutorial-assessment-vmware

✓ This tutorial requires a VMware environment. If you do not have that environment check out the next practice which uses Microsoft Online Labs.

✓ Now might be a good time to review the FAQ at the reference link to ensure you learn all the finer points about Azure Migrate.

For more information, you can see:

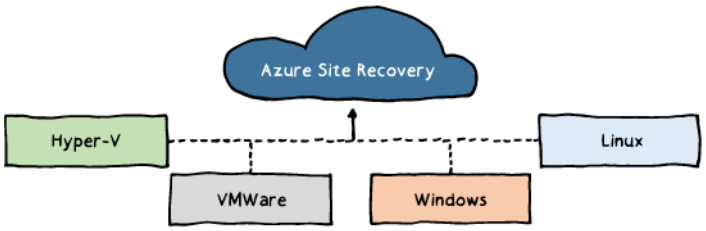
Azure Migrate - Frequently Asked Questions (FAQ) - **https://docs.microsoft.com/en-us/azure/migrate/resources-faq**

Overview of Azure Site Recovery (ASR)

Video: ASR Capabilities

ASR Scenarios

You can use Azure Site Recovery to replicate on-premises physical or virtual machines running Windows or Linux. Azure Site Recovery includes support for both Hyper-V and VMware virtual machines. You can replicate data from your on-premises datacenter to Azure or to a secondary site. Orchestration is built in with Azure Site Recovery, which means that the management of replication, failover, and recovery is included. For example, should a virtual machine or service fail in your datacenter, you can use Azure Site Recovery to failover to the replicated resource in either Azure or your secondary site.



Azure Site Recovery works in the following three scenarios:

- **Hyper-V Virtual Machine Replication**. When Virtual Machine Manager (VMM) is used to manage Hyper-V virtual machines, you can use Azure Site Recovery to replicate them to Azure or to a secondary datacenter. If you do not use VMM to manage your virtual machines, you can use Azure Site Recovery to replicate them to Azure only.
- **VMware Virtual Machine Replication**. You can perform the replication of virtual machines by VMware to a secondary site that is also running VMware. You also can replicate to Azure.
- **Physical Windows and Linux machines**. You can replicate physical machines running either Windows or Linux to a secondary site or to Azure.

✓ Are you considering using Azure Site Recovery?

For more information, you can see:

Site recovery - **https://azure.microsoft.com/en-us/services/site-recovery/**

ASR Features

Here are some reasons to use Azure Site Recovery.

- **Eliminate the need for disaster recovery sites**. Your environment can be protected by automating the replication of the virtual machines based on policies that you set and control. Site Recovery is heterogeneous and can protect Hyper-V, VMware, and physical servers.
- **Reduce infrastructure costs**. Lower your on-premises infrastructure costs by using Azure as a secondary site for conducting business during outages. Or, eliminate datacenter costs altogether by moving to Azure and setting up disaster recovery between Azure regions. You can pre-assess network, storage, and compute resources needed to replicate applications from on-premises to Azure—and pay only for compute and storage resources needed to run apps in Azure during outages.
- **Automatically replicate to Azure**. Automate the orderly recovery of services in the event of a site outage at the primary datacenter. Automate the orderly recovery of services in the event of a site outage at the primary datacenter.
- **Safeguard against outages of complex workloads**. Protect applications in SQL Server, SharePoint, SAP, and Oracle.
- **Extend or boost capacity**. Applications can be Migrated to Azure with just a few clicks or burst to Azure temporarily when you encounter a surge in demand.
- **Continuous health monitoring**. Site Recovery monitors the state of your protected instances continuously and remotely from Azure. When replicating between two sites you control, your virtual machines' data and replication remains on your networks. All communication with Azure is encrypted.

✓ Are you interested in any of these specific features? Which one is most important to you?

For more information, you can see:

Site Recovery - **https://azure.microsoft.com/en-us/services/site-recovery/**

Site Recovery Pricing - **https://azure.microsoft.com/en-us/pricing/details/site-recovery/**

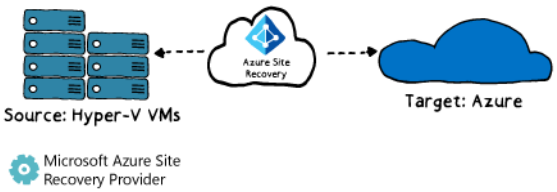
Video: ASR Features

Video: ASR End to End

Using ASR: A Look Ahead

We've talked about the concepts behind ASR, ASR features and typical scenarios for its use. This topic serves as an introduction to the next 2 lessons, where we'll look at using ASR to migrate Hyper-V VMs – that are not being managed by System Center Virtual Machine Manager – to Azure. In this scenario you replicate on-premises Hyper-V VMs to Azure storage. Then, you fail over from on-premises to Azure. After failover, your apps and workloads are available and running on Azure VMs.

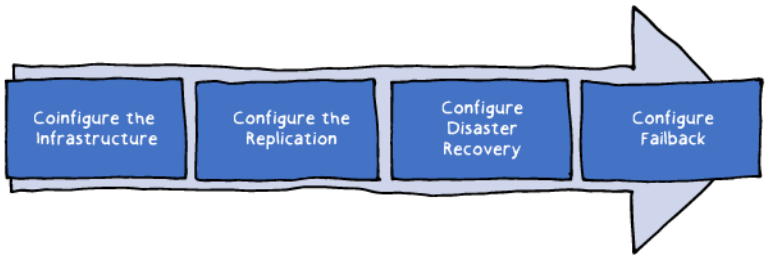
The basic architecture of this scenario looks like this. The following diagram shows a set of on-premises Hyper-V VMs as the source from which those VMs are then replicated using ASR to the target (Azure).



The Process

Migrating Hyper-V VMs to Azure with ASR involves four basic steps: Configure the Infrastructure, Configure the Replication, Configure Disaster Recovery, and Configure Failback. The next two lessons will go into detail on the entire process and there will be a video for each step.

The next lesson, Preparing the Infrastructure, covers step 1, the infrastructure configuration. The subsequent lesson, Completing the Migration Process, covers the remaining steps in the migration process.



For more information, you can see:

Hyper-V to Azure replication architecture - **https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-azure-architecture**

Preparing the Infrastructure

Configure Azure

In these next two lessons, we will focus on migrating Hyper-V virtual machines to Azure. No matter the type of virtual machines you are migrating to Azure there are certain configuration tasks that you will always perform. These tasks are completed in Azure and focus on permissions, storage, and networking.

- **Verify account permissions**. If you are the subscription owner then you have the appropriate permissions. If you are delegating permissions your account should be assigned the Virtual Machine Contributor built-in role. In addition, to manage Site Recovery operations in a vault, your account should be assigned the Site Recovery Contributor build-in role.
- **Create a storage account**. Replicated data from on-premises VM workloads is stored in the storage account. Azure VMs are created with the replicated data when failover from your on-premises site occurs. This should be a general purpose storage, not blob storage. Keep the default Read-access geo-redundant storage for storage redundancy. Replicated data in Azure storage is also encrypted so leave Secure transfer required as Disabled.
- **Create a virtual network**. VM networks are mapped to Azure virtual networks. When Azure VMs are created after failover, they are added to this Azure network. You should also think about intersite connectivity. Your data must move between your on-premises datacenter and Azure. You will need good bandwidth for that.

For more information, you can see:

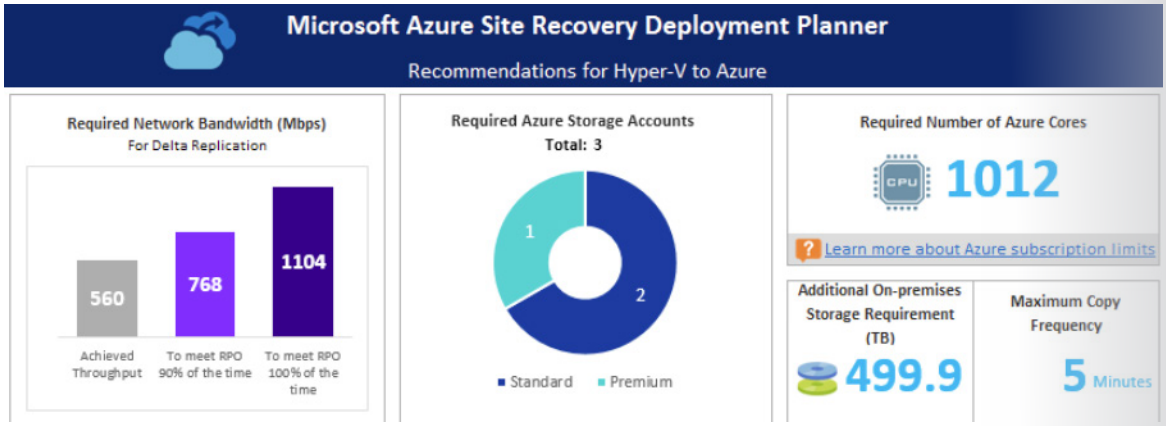
Prepare Azure - **https://docs.microsoft.com/en-us/azure/site-recovery/tutorial-prepare-azure#verify-account-permissions**[16]

Azure Storage - **https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-azure-support-matrix#azure-storage**[17]

ASR Deployment Planner

To help you make good decisions on storage and networking you can use the Azure Site Recovery deployment planner. This is a command-line tool that remotely profiles your Hyper-V VMs to gives insight to bandwidth and Azure storage requirements. Running this tool is critical to ensuring you correctly configure your new infrastructure.

16 https://docs.microsoft.com/en-us/azure/site-recovery/tutorial-prepare-azure
17 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-azure-support-matrix



The generated Microsoft Excel report contains the following information:

- **On-premises summary**[18]
- **Recommendations**[19]
- **VM storage placement**[20]
- **Compatible VMs**[21]
- **Incompatible VMs**[22]
- **On-premises storage requirement**[23]
- **Initial Replication batching**[24]
- **Cost estimation**[25]

✓ You can run this tool without installing any Azure Site Recovery components on-premises. However, to get accurate achieved throughput results, we recommend that you run the planner on a Windows Server that has the same hardware configuration as that of one of the Hyper-V servers that you will use to enable disaster recovery protection to Azure.

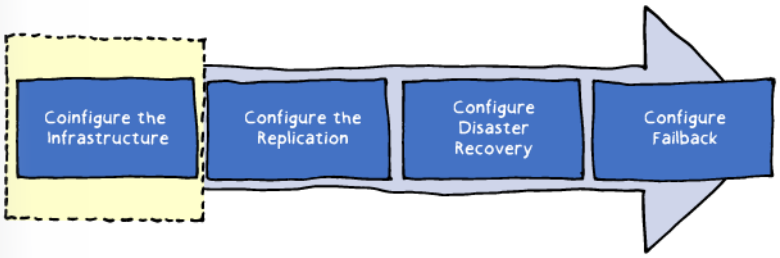
For more information, you can see:

Site Recovery Deployment Planner for Hyper-V to Azure - **https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-overview**

Analyze the Azure Site Recovery Deployment Planner Report - **https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-analyze-report#vm-storage-placement-recommendation**[26]

18 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-analyze-report
19 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-analyze-report
20 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-analyze-report
21 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-analyze-report
22 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-analyze-report
23 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-analyze-report
24 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-analyze-report
25 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-cost-estimation
26 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-deployment-planner-analyze-report

Configure the Infrastructure: Requirements



Let's begin with the first step, Configure the Infrastructure. This step ensures all the prerequisites are met so you will have a successful migration. Here are a few of the most important requirements.

- **Verify Hyper-V operating system requirements**[27]. Windows Server 2016 (including server core installation), Windows Server 2012 R2 with latest updates are supported. Mixing hosts running Windows Server 2016 and 2012 R2 isn't supported.
- **Verify Hyper-V host and guest storage requirements**[28]. Hyper-V host storage can include SMB 3.0, SAN (iSCSI), and Multi-path (MPIO). Notice at this reference link that Hyper-V VM guest storage has some limitations. For example, shared cluster disks and encrypted disks are not supported.
- **Verify internet access**[29]. The simplest configuration is for the Hyper-V hosts to have direct access to the internet without using a proxy. This reference link provides test URLs to ensure connectivity for access control, replication, and telemetry.
- Prepare VMs so that you can access them after failover (optional). During a failover scenario you may want to connect to your replicated on-premises network. If you plan to connect to Windows VMs using RDP after failover make sure that TCP, and UDP rules are added for the Public profile, and that RDP is allowed in Windows.

✓ The reference link shows how to troubleshoot Windows RDP connections and has more details on how to configure the firewall rules.

For more information, you can see:

Prepare on-premises Hyper-V servers for disaster recovery to Azure - **https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-prepare-on-premises-tutorial**

Troubleshooting remote desktop connection after failover using ASR - **https://social.technet.microsoft.com/wiki/contents/articles/31666.troubleshooting-remote-desktop-connection-after-failover-using-asr.aspx**

Configure the Infrastructure: Process

The Azure portal will step you through configuring the infrastructure for your migration.

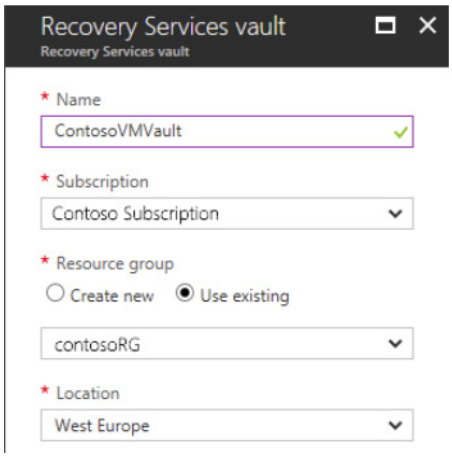
27 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-azure-support-matrix
28 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-azure-support-matrix
29 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-prepare-on-premises-tutorial

1. **Protection goal**. In this lesson we are migrating on-premises Hyper-V machines to Azure. We are not using System Center VMM.
2. **Source**. To set up the source environment, you create a Hyper-V site, and add Hyper-V hosts to the site. Then you download and install the Azure Site Recovery Provider agent on each host and register the Hyper-V site in the vault.
3. **Target**. Select the subscription and the resource group where the Azure VMs will be created after failover.
4. **Replication settings**. Select your replication settings.
5. **Deployment planning**. You will be asked to confirm you have conducted deployment planning for your migration. This could include the ASR Deployment Planner.

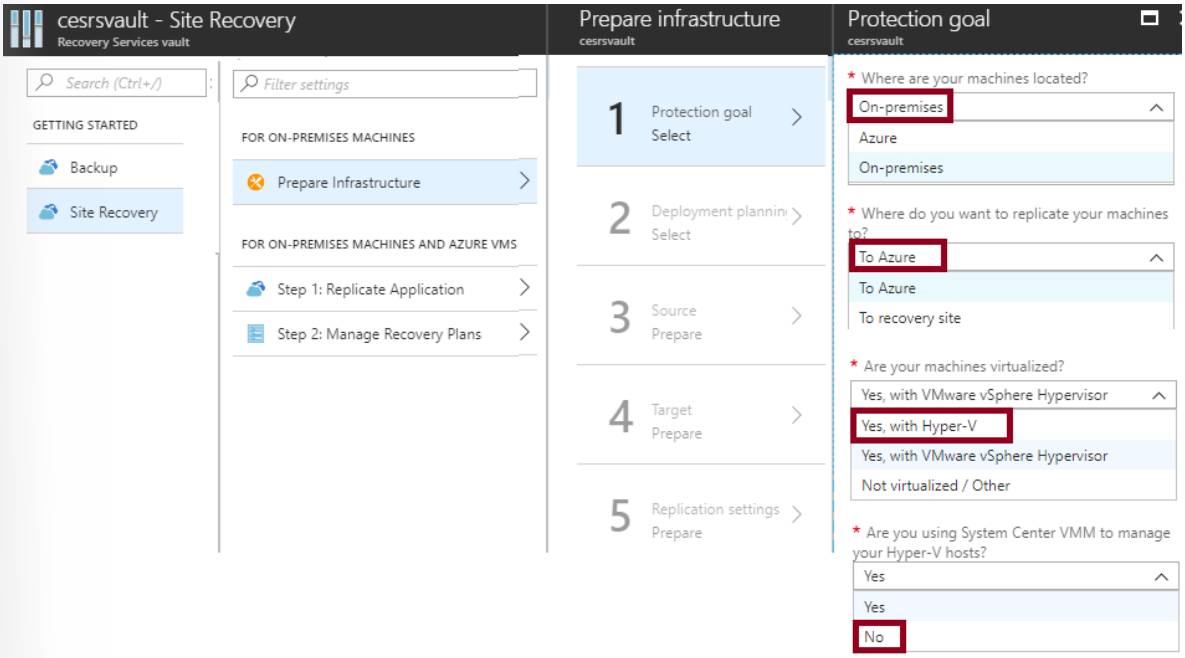
✓ You will run the Provider setup file (AzureSiteRecoveryProvider.exe) on each Hyper-V host. It can take up to 30 minutes for VMs to appear in Azure.

Recovery Services Vault

You will need to create a recovery services vault in the portal. You can do this from the Backup and Site Recovery (OMS) link.



Once your recovery services vault is created you can use the Site Recovery blade. Notice that you must first prepare the infrastructure and select a protection goal. In this example, our machines are located on-premises, we are replicating our machines to Azure, the machines are virtualized with Hyper-V, and we are not using System VMM to manage the Hyper-V hosts. The next steps are customized to your choices.



For more information, you can see:
Recovery Services vaults overview - **https://docs.microsoft.com/en-us/azure/backup/backup-azure-recovery-services-vault-overview**

Video - Configure the Infrastructure

In this first video of a 4-part video series dedicated to Hyper-V, you walk through the process of setting up your source and target infrastructure and creating replication policies that meet the replicate and recovery service level objectives of your organization.

Practice: Preparing the Azure Environment

ASR helps keep your business apps up and running during planned and unplanned outages. ASR manages and orchestrates disaster recovery of on-premises machines and Azure virtual machines (VMs), including replication, failover, and recovery.

This lesson incorporate a series of tutorials. This **tutorial**[30] is the first in the series, and it teaches you how to prepare Azure components for disaster recovery for on-premises VMs. This preparation applies whether you're protecting on-premises VMware VMs, Hyper-V VMs, or physical servers.

In this tutorial, you learn how to:

- Verify that your Azure account has replication permissions.
- Create an Azure storage account. Images of replicated machines are stored in it.
- Create a Recovery Services vault. A vault holds metadata and configuration information for VMs, and other replication components.
- Set up an Azure network. When Azure VMs are created after failover, they're joined to this Azure network.

✓ The tutorials in this lesson assume you have an existing set of on-premises resources to migrate to Azure.

For more information, you can see:

Prepare Azure resources for replication of on-premises machines – **https://docs.microsoft.com/en-us/azure/site-recovery/tutorial-prepare-azure**

Practice: Preparing On-premises Hyper-V Servers

This second **tutorial**[31] shows you how to prepare your on-premises Hyper-V infrastructure for the purpose of replicating Hyper-V VMs to Azure. You have the choice to perform this using System Center VMM, but that is optional. However, the next tutorial is relevant to Hyper-V VMs that are not managed by System Center VMM, so you may want to keep that in mind.

In this tutorial, you learn how to:

- Review Hyper-V requirements, and VMM requirements if applicable.
- Prepare VMM if applicable
- Verify internet access to Azure locations
- Prepare VMs so that you can access them after failover to Azure

✓ The tutorials in this lesson assume you have an existing set of on-premises resources to migrate to Azure. In the next lesson, you'll have an opportunity to complete the migration process.

For more information, you can see:

Prepare Azure resources for replication of on-premises machines – **https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-prepare-on-premises-tutorial**

30 https://docs.microsoft.com/en-us/azure/site-recovery/tutorial-prepare-azure
31 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-prepare-on-premises-tutorial

Completing the Migration Process

Video - Configure Failback

In this final video of our 4-part video series dedicated to Hyper-V, we walk you through the process of replicating your virtual machines running in Azure back to your on-premises datacenter to perform a failback.

Troubleshoot Hyper-V to Azure Failover

In this second video of our 4-part video series dedicated to Hyper-V, we walk you through the process of protecting and replicating your Hyper-V virtual machines to Azure. Premium storage is now supported for VMware VM, Hyper-V VM, and physical server replication.

Customize the Recovery Plan

This third **tutorial**[32] in the series shows you how to Set up disaster recovery of on-premises Hyper-V VMs to Azure migrate on-premises VMs and physical servers to Azure. This is all part of the process of using ASR to manage your migration and disaster recovery strategy and your replication orchestration to create failover and failback of your on-premises and Azure VMs.

In this tutorial, you learn how to:

- Select a replication goal
- Set up the source and target environment

32 https://docs.microsoft.com/en-us/azure/site-recovery/migrate-tutorial-on-premises-azure

- Set up a replication policy
- Enable replication
- Run a test migration to make sure everything's working as expected
- Run a one-time failover to Azure

✓ The tutorials in this lesson assume you have an existing set of on-premises resources to migrate to Azure. This tutorial assumes that you have already completed the tasks in the previous lesson's tutorials.

For more information, you can see:

Migrate on-premises machines to Azure – **https://docs.microsoft.com/en-us/azure/site-recovery/migrate-tutorial-on-premises-azure**

Practice: Using PowerShell to Migrate VMs to Azure

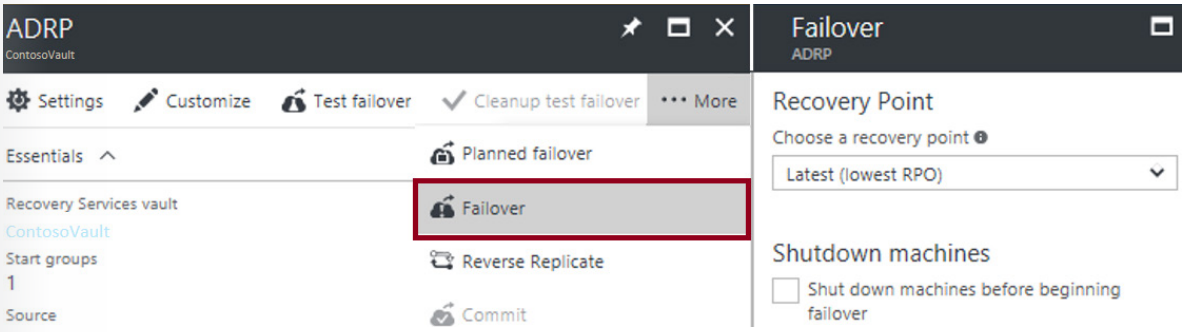
In this third video of our 4-part video series dedicated to Hyper-V, we walk you through the process of creating a Recovery Plan, performing non-disruptive disaster recover drill with Test Failover, and performing Planned and Unplanned Failover to Azure.

Planned and Unplanned Failovers

There are two types of failover: planned and unplanned. An example of a planned failover is when your datacenter has scheduled downtime. An example of an unplanned failover is when your datacenter has a power outage.

Scenario	Requirement
Planned failover due to an upcoming datacenter downtime	Zero data loss for the application when a planned activity is performed
Failover due to an unplanned datacenter downtime (natural or IT disaster)	Minimal data loss for the application

Failover isn't automatic. You can initiate failovers in the portal or with PowerShell, but it is a deliberate act.



✓ Never cancel a failover in progress. Also, we recommend LRS or GRS storage, so the data is resilient if a regional outage occurs, or if the primary region can't be recovered.

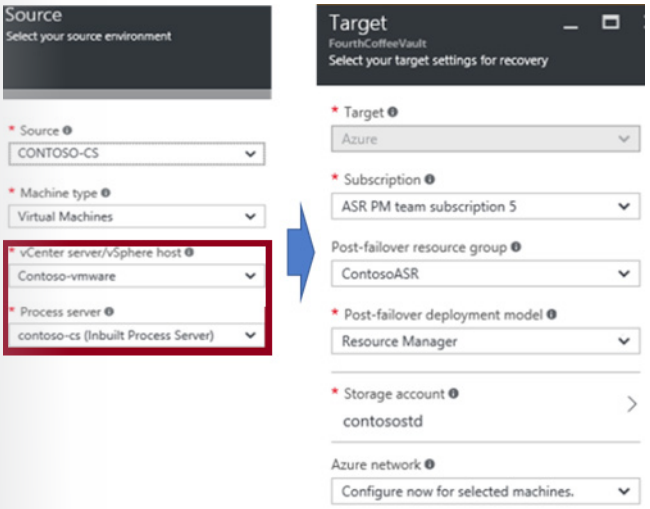
For more information, you can see:

Failover in Site Recovery - **https://docs.microsoft.com/en-us/azure/site-recovery/site-recovery-failover**

Create and customize recovery plans - **https://docs.microsoft.com/en-us/azure/site-recovery/site-recovery-create-recovery-plans**

Enable Replication

To enable replication on your virtual machines, begin by configuring the source and the target. In the example below, the source for the VMs is a VMware vCenter server. In addition, the process server is selected. For the target, the Azure subscription is selected, as well as the resource group, storage account, and network information.



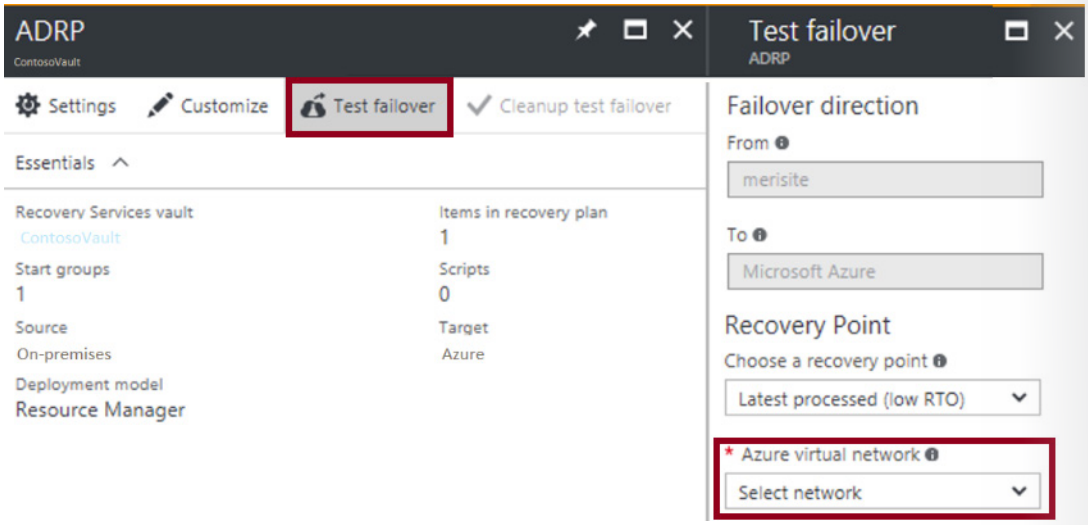
For more information, you can see:

Enable replication - **https://docs.microsoft.com/en-us/azure/site-recovery/vmware-azure-enable-replication**

Test Failover

We recommend you periodically perform a test failover to test your ASR plan. A planned test failover will validate your replication and disaster recovery strategy, without any data loss or downtime. A test failover

doesn't impact ongoing replication, or your production environment. You can run a test failover on a specific virtual machine (VM), or on a recovery plan containing multiple VMs.



Here are some best practices:

- Failover to a network separate from your production network. Failover to your production network can lead to unexpected consequences, such as two virtual machines with the same identity.
- Create failover groups of virtual machines, this will make it easier to manage large numbers of machines. For example, you could have different groups for the frontend, database, and backend tiers.
- Consider integrating with Azure Automation to help you extend your recovery plans. For example, adding a public IP address to your virtual machine.

✓ After a test failover be sure to complete the test so the virtual machines are removed.

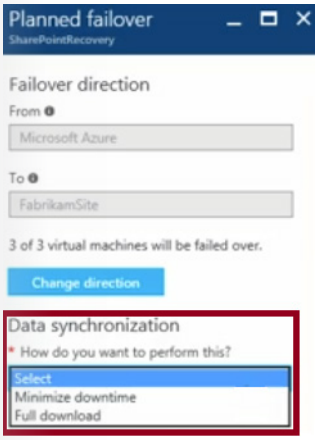
For more information, you can see:

Test failover to Azure in Site Recovery - **https://docs.microsoft.com/en-us/azure/site-recovery/site-recovery-test-failover-to-azure**

Add Azure Automation runbooks to recovery plans - **https://docs.microsoft.com/en-us/azure/site-recovery/site-recovery-runbook-automation**

Failback

Failback operation is a planned failover from Azure to the on-premises site. You will verify the failover direction (from Azure) and select the source and target locations. There are two ways to synchronize the data: Full download and Minimize downtime.



- **Full download**. A full download is faster but requires the VM to be shutdown. This results in more downtime. Choose this option if there have been a lot of changes to the VM or it has been running more than one month.
- **Minimize downtime**. This option takes more time to synchronize the changes, but the VM is not shut down. So, there is less downtime.

After the download completes you should log on to the on-premises VM to check it's available as expected.

The on-premises VM is now in a Commit Pending state. Click Commit. It deletes the Azure VMs and its disks and prepares the on-premises VM for reverse replication. To start replicating the on-premises VM to Azure, enable Reverse Replicate. It triggers replication of delta changes that have occurred since the Azure VM was switched off.

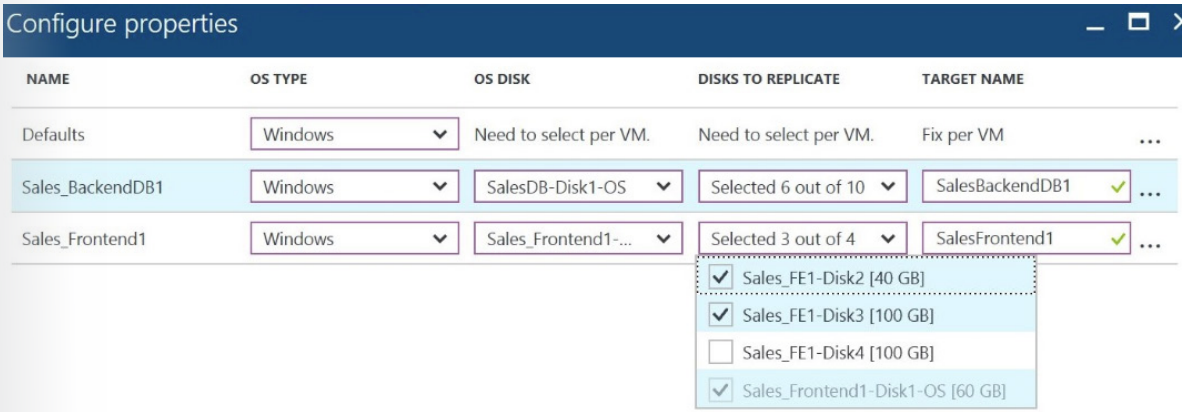
✓ Failback does have an option to create the virtual machine if it does not exist. This means you can create the virtual machine on a new host.

For more information, you can see:

Run a failback for Hyper-V VMs - **https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-azure-failback**

Virtual Machine Replication

Replication involves selecting your virtual machines and specifying the OS type, OS disk, disks to replicate, and virtual machine target name.



Excluding disks

Initially, you may think it is important to select all data disks for replication. However, you can save storage and networking resources by excluding disks for the data that is not important or doesn't need to be replicated. For example, paging files.

Paging files have a lot of churn and will generate a lot of replication events. So, you can use the following steps to optimize replication of a virtual machine with a single virtual disk that has both the operating system and the paging file:

1. Split the single virtual disk into two virtual disks. One virtual disk has the operating system, and the other has the paging file.
2. Exclude the paging file disk from replication.

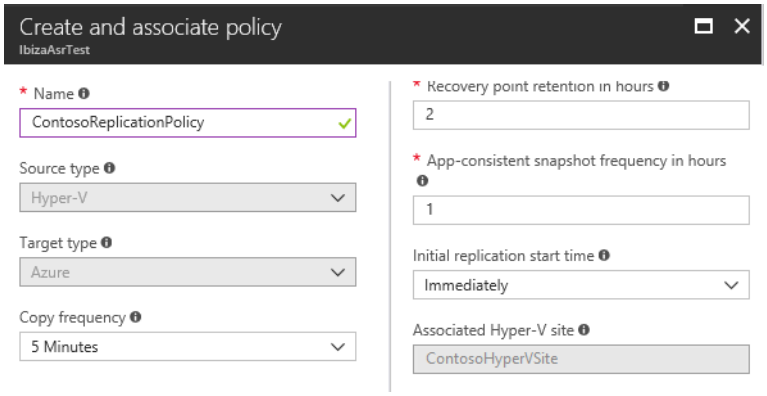
✓ Can you think of any other examples where splitting disks could save replication time? The reference link discusses SQL server tempdb files.

For more information, you can see:

Excluding disks to replicate - **https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-exclude-disk**

Replication Policy

After selecting what you want to replicate you will create a replication policy. Pay attention to the copy frequency, recovery point retention, app-consistent snapshot, and initial replication start time settings. You can create different replication scenarios.



- **Copy frequency**. Specify how often you want to replicate delta data after the initial replication.
- **Recovery point retention**. Specify in hours how long the retention window will be for each recovery point.
- **App-consistent snapshot**. Specify the frequency that recovery points containing app-consistent snapshots will be created. Hyper-V application-consistent snapshot takes a point-in-time snapshot of the application data inside the virtual machine.
- **Initial replication start time**. Specify when to start the initial replication. The replication occurs over your internet bandwidth, so you might want to schedule it outside your busy hours.

✓ For Hyper-V to Azure replication policies, the 15-minute copy frequency option is being retired in favor of the 5-minute, and 30-second copy frequency settings. Replication policies using a 15-minute copy frequency will automatically be updated to use the 5-minute copy frequency setting.

For more information, you can see:

Set up a replication policy - **https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-azure-tutorial#set-up-a-replication-policy**[33]

33 https://docs.microsoft.com/en-us/azure/site-recovery/hyper-v-azure-tutorial

Online Lab - Evaluating and Performing Server Migration to Azure

Lab Steps

Online Lab - Evaluating and Performing Server Migration to Azure

Topic: Implementing Azure to Azure migration

Scenario

Adatum Corporation wants to migrate their existing Azure VMs to another region

Objectives

After completing this lab, you will be able to:

- Implement Azure Site Recovery Vault
- Migrate Azure VMs between Azure regions

Lab Setup

Estimated Time: 45 minutes

User Name: **Student**

Password: **Pa55w.rd**

Exercise 1: Implement prerequisites for migration of Azure VMs by using Azure Site Recovery

The main tasks for this exercise are as follows:

1. Deploy an Azure VM to be migrated
2. Create an Azure Recovery Services vault

Task 1: Deploy an Azure VM to be migrated

1. From the lab virtual machine, start Microsoft Edge and browse to the Azure portal at **http://portal.azure.com** and sign in by using the Microsoft account that has the Owner role in the target Azure subscription.
2. In the Azure portal, in the Microsoft Edge window, start a **PowerShell** session within the **Cloud Shell**.
3. If you are presented with the **You have no storage mounted** message, configure storage using the following settings:
    - Subsciption: the name of the target Azure subscription

- Cloud Shell region: the name of the Azure region that is available in your subscription and which is closest to the lab location
- Resource group: the name of a new resource group **az3000600-LabRG**
- Storage account: a name of a new storage account
- File share: a name of a new file share

4. From the Cloud Shell pane, create a resource group by running (replace the `<Azure region>` placeholder with the name of the Azure region that is available in your subscription and which is closest to the lab location)

```
New-AzureRmResourceGroup -Name az3000601-LabRG -Location <Azure region>
```

1. From the Cloud Shell pane, upload the Azure Resource Manager template **D:\LabFiles\06\azuredeploy06.json** into the home directory.
2. From the Cloud Shell pane, upload the parameter file **D:\LabFiles\06\azuredeploy06.parameters.json** into the home directory.
3. From the Cloud Shell pane, deploy an Azure VM hosting Windows Server 2016 Datacenter by running:

```
New-AzureRmResourceGroupDeployment -ResourceGroupName az3000601-LabRG -TemplateFile azuredeploy06.json -TemplateParameterFile azuredeploy06.parameters.json
```

> **Note:** Do not wait for the deployment to complete but instead proceed to the next task.

Task 2: Implement an Azure Site Recovery vault

1. From Azure Portal, create an instance of **Backup and Site Recovery (OMS)** Recovery Services vault with the following settings:
   - Name: **vaultaz3000602**
   - Subscription: the name of the target Azure subscription
   - Resource group: the name of a new resource group **az3000602-LabRG**
   - Location: the name of an Azure region that is available in your subscription and which is different from the region you deployed an Azure VM in the previous task
2. Wait until the vault is provisioned. This will take about a minute.

Result: After you completed this exercise, you have created an Azure VM to be migrated and an Azure Site Recovery vault that will host the migrated disk files of the Azure VM.

Exercise 2: Migrate an Azure VM between Azure regions by using Azure Site Recovery

The main tasks for this exercise are as follows:

1. Configure Azure VM replication
2. Review Azure VM replication settings
3. Disable replication of an Azure VM and delete the Azure Recovery Services vault

Task 1: Configure Azure VM replication

1. In the the Azure portal, navigate to the blade of the newly provisioned Azure Recovery Services vault.
2. Set the protection goal to **Azure VMs to Azure**.
3. Enable replication by specifying the following settings:
   - Source: **Azure**
   - Source location: the same Azure region into which you deployed the Azure VM in the previous exercise of this lab
   - Azure virtual machine deployment model: **Resource Manager**
   - Source resource group: **az3000601-LabRG**
   - Virtual machines: **az300061-vm**
   - Target location: the name of an Azure region that is available in your subscription and which is different from the region you deployed an Azure VM in the previous task
   - Target resource group: **(new) az3000601-LabRG-asr**
   - Target virtual network: **(new) az3000601-vnet-asr**
   - Cache storage account: accept the default setting
   - Replica managed disks: **(new) 1 premium disk(s), 0 standard disk(s)**
   - Target availability sets: **Not Applicable**
   - Replication policy: **Create new**
   - Name: **12-hour-retention-policy**
   - Recovery point retention: **12 Hours**
   - App consistent snapshot frequency: **6 Hours**
   - Multi-VM consistency: **No**
4. Initiate creation of target resources.
5. Enable the replication.
6. **Note**: Wait for the operation of enabling the replication to complete. Then proceed to the next task.

Task 2: Review Azure VM replication settings

1. In the Azure portal, from the Azure Site Recovery vault blade, navigate to the replicated item blade representing the Azure VM **az3000601-vm**.
2. On the replicated item blade, review the **Health and status**, **Latest available recovery points**, and **Failover readiness** sections. Note the **Failover** and **Test Failover** entries in the toolbar. Scroll down to the **Infrastructure view**.
3. If time permits, wait until the status of the Azure VM changes to **Protected**. This might take additional 15-20 minutes. At that point, examine the values **Crash-consistent** and **App-consistent** recovery points. In order to view **RPO**, you should perform a test failover.

Task 3: Disable replication of an Azure VM and delete the Azure Recovery Services vault

1. In the Azure portal, disable replication of the Azure VM **az3000601-vm**.
2. Wait until the replication is disabled.
3. From the Azure portal, delete the Recovery Services vault.
4. **Note**: You must ensure that the replicated item is removed first before you can delete the vault.

Result: After you completed this exercise, you have implemented automatic replication of an Azure VM.

Review Questions

Module 1 Review Questions

Migration Goals

You are hired by an organization to evaluate IT department regarding needs and department spending.

The organization uses a pre-purchase model. Resources have a three to five year life cycle. All hardware and software is managed and maintained on-premises or in shared datacenters.

You need to evaluate whether to migrate some or all resources to Azure

What should you consider? What are the benefits? Which tool can help you plan and assess costs and cost-savings?

Suggested Answer ↓

When migrating any workload to a new environment, whether it be another datacenter or to a public cloud, you should have a clear set of goals for migration in mind. Note that there are both technology-focused and business-focused goals that motivate potential migrations. All migration efforts should result in direct benefits to the organization's business.

Azure Site Recovery

You are hired to recommend a disaster recovery solution for an organization. The organization has a mix of Hyper-V virtual machines (VMs), VMware VMs, and physical servers.

You are considering Azure Site Recovery.

What must be taken into consideration for each type of resource? Why is it important that Azure Site Recovery includes Orchestration? What are some benefits to consider for Azure Site Recovery and Orchestration?

Suggested Answer ↓

You can use Azure Site Recovery to replicate on-premises physical or virtual machines that run Windows or Linux. Azure Site Recovery includes support for Hyper-V and VMware virtual machines. You can replicate data from your on-premises datacenter to Azure or to a secondary site.

You can use Azure Site Recovery to failover to the replicated resource in either Azure or your secondary site.

Here are some reasons to use Azure Site Recovery.

• Eliminate the need for disaster recovery sites. Your environment can be protected by automating the replication of the virtual machines based on policies that you set and control. Site Recovery is heterogeneous and can protect Hyper-V, VMware, and physical servers.
• Reduce infrastructure costs. Lower your on-premises infrastructure costs by using Azure as a secondary site for conducting business during outages. Or, eliminate datacenter costs altogether by moving to Azure and setting up disaster recovery between Azure regions. You can pre-assess network, storage, and compute resources needed to replicate applications from on-premises to Azure—and pay only for compute and storage resources needed to run apps in Azure during outages.

• Automatically replicate to Azure. Automate the orderly recovery of services in the event of a site outage at the primary datacenter. Automate the orderly recovery of services in the event of a site outage at the primary datacenter.
• Safeguard against outages of complex workloads. Protect applications in SQL Server, SharePoint, SAP, and Oracle.
• Extend or boost capacity. Applications can be Migrated to Azure with just a few clicks or burst to Azure temporarily when you encounter a surge in demand.
• Monitor resource health. Site Recovery monitors the state of your protected instances continuously and remotely from Azure. When replicating between two sites you control, your virtual machines' data and replication remains on your networks. All communication with Azure is encrypted.

Planned and Unplanned Failovers

An organization plans to replicate all on-premises servers and systems to Azure as a failover location in the event that the on-premises datacenter becomes unavailable.

How do you initiate a failover? What types of failover scenarios should you consider? How can you ensure data resiliency?

Suggested Answer ↓

Failover isn't automatic. You can initiate failovers in the portal or with PowerShell, but it is a deliberate act.

There are two types of failover: planned and unplanned. An example of a planned failover is when your datacenter has scheduled downtime. An example of an unplanned failover is when your datacenter has a power outage.

LRS or GRS storage is recommended, so the data is resilient if a regional outage occurs, or if the primary region cannot be recovered.

Module 2 Module Implementing and Managing Application Services

Deploying Web Apps

Web Apps Features

In App Service, a web app is the compute resources that Azure provides for hosting a website or web application.

The compute resources may be on shared or dedicated virtual machines (VMs), depending on the pricing tier that you choose. Your application code runs in a managed VM that is isolated from other customers.

Here are some key features and capabilities of App Service:

- **Multiple languages and frameworks** - App Service has first-class support for ASP.NET, Node.js, Java, PHP, and Python. You can also run Windows PowerShell and other scripts or executables on App Service VMs.
- **DevOps optimization** - Set up continuous integration and deployment with Visual Studio Team Services, GitHub, or BitBucket. Promote updates through test and staging environments. Perform A/B testing. Manage your apps in App Service by using Azure PowerShell or the cross-platform command-line interface (CLI).
- **Global scale with high availability** - Scale up or out manually or automatically. Host your apps anywhere in Microsoft's global datacenter infrastructure, and the App Service SLA promises high availability.
- **Connections to SaaS platforms and on-premises data** - Choose from more than 50 connectors for enterprise systems (such as SAP, Siebel, and Oracle), SaaS services (such as Salesforce and Office 365), and internet services (such as Facebook and Twitter). Access on-premises data using Hybrid Connections and Azure Virtual Networks.
- **Security and compliance** - App Service is ISO, SOC, and PCI compliant.
- **Application templates** - Choose from an extensive list of templates in the Azure Marketplace that let you use a wizard to install popular open-source software such as WordPress, Joomla, and Drupal.

- **Visual Studio integration** - Dedicated tools in Visual Studio streamline the work of creating, deploying, and debugging.
- **API and mobile features** - Turn-key CORS support for RESTful API scenarios, authentication for mobile app scenarios, and offline data sync, and push notifications.
- **Serverless code** - Run a code snippet or script on-demand without having to explicitly provision or manage infrastructure, paying only for the compute time that the code uses.

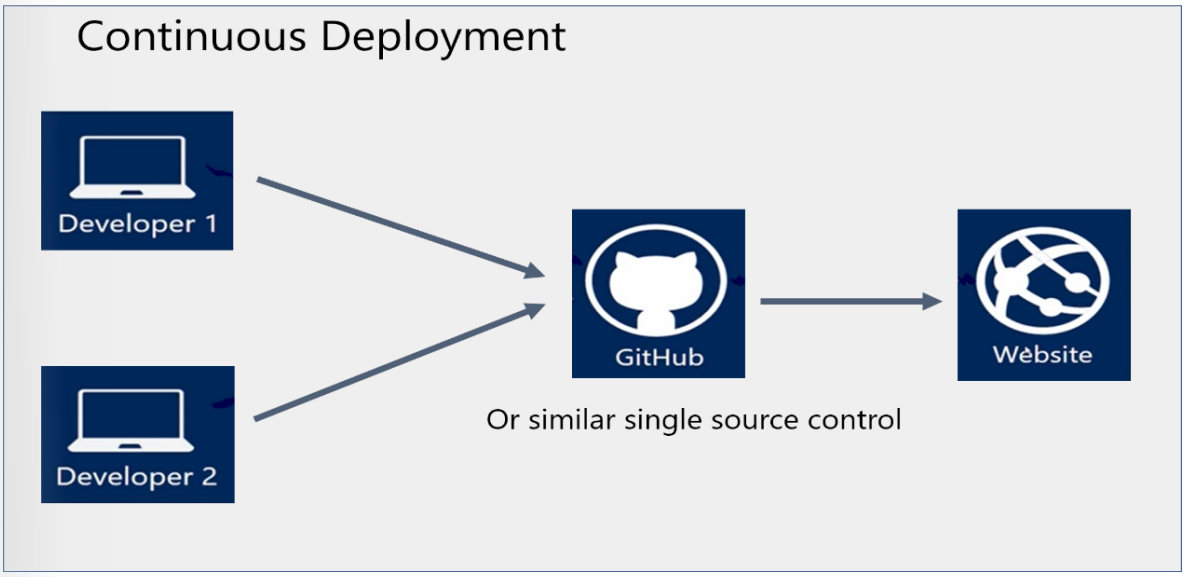
✓ Do you have any web app projects in mind for your organization?

For more information, you can see:

Web apps overview - **https://docs.microsoft.com/en-us/azure/app-service/app-service-web-overview**

What is Continuous Development?

Multiple developers want to be able to work in a single source control. Whenever code updates are pushed to the source control, then the website or web app will automatically pick up the updates. A continuous deployment workflow publishes the most recent updates from a project.



The Azure portal makes it easy to set up continuous deployment from GitHub, Bitbucket, or Visual Studio Team Services. You can also set up continuous deployment from a git host that the portal doesn't directly support, like GitLab.

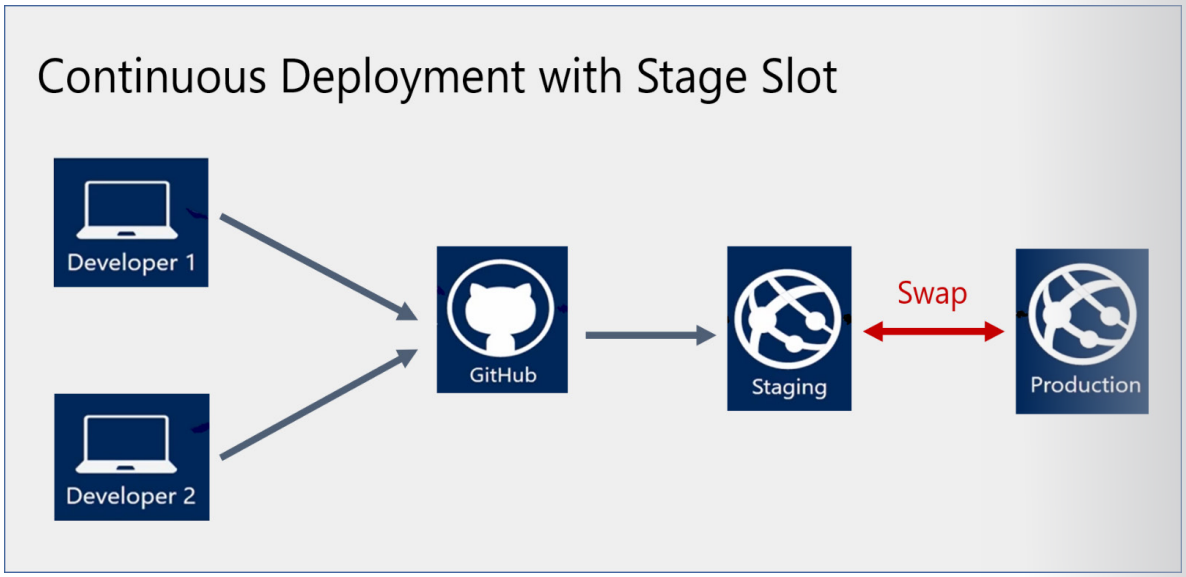
✓ Continuous deployment is a great option for projects where multiple and frequent contributions are being integrated.

For more information , you can see:

Continuous deployment - **https://github.com/projectkudu/kudu/wiki/Continuous-deployment#setting-up-continuous-deployment-using-manual-steps**[1]

1 https://github.com/projectkudu/kudu/wiki/Continuous-deployment

Staging Environments in App Service



When you deploy your web app, mobile back end, and API app to App Service, you can deploy to a separate deployment slot instead of the default production slot when running in the **Standard** or **Premium** App Service plan mode.

Deployment slots are live apps with their own hostnames. App content and configurations elements can be swapped between two deployment slots, including the production slot. Using separate staging and production slots has several advantages.

- You can validate app changes in a staging deployment slot before swapping it with the production slot.
- Deploying an app to a slot first and swapping it into production ensures that all instances of the slot are warmed up before being swapped into production. This eliminates downtime when you deploy your app. The traffic redirection is seamless, and no requests are dropped because of swap operations. This entire workflow can be automated by configuring Auto Swap when pre-swap validation is not needed.
- After a swap, the slot with previously staged app now has the previous production app. If the changes swapped into the production slot are not as you expected, you can perform the same swap immediately to get your "last known good site" back.

✓ Each App Service plan mode supports a different number of deployment slots. To find out the number of slots your app's mode supports, see **App Service Limits**[2].

For more information, you can see:

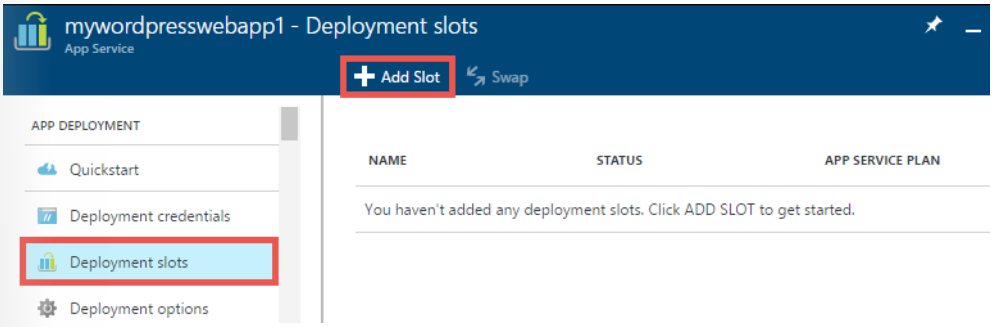
Set up staging environments - **https://docs.microsoft.com/en-us/azure/app-service/web-sites-staged-publishing?toc=%2Fazure%2Fapp-service%2Ftoc.json#add-a-deployment-slot**[3]

App Service Web App – block web access to non-production deployment slots - **http://ruslany.net/2014/04/azure-web-sites-block-web-access-to-non-production-deployment-slots/**

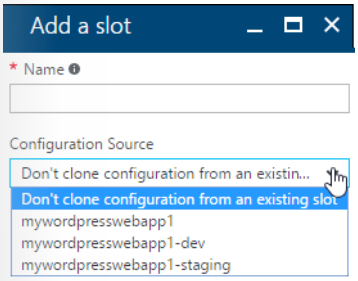
2 https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits
3 https://docs.microsoft.com/en-us/azure/app-service/web-sites-staged-publishing?toc=%2Fazure%2Fapp-service%2Ftoc.json

Add a Deployment Slot

Your app must be running in the **Standard** or **Premium** tier for you to enable multiple deployment slots. If the app is not already in the Standard or Premium tier, you will receive a message indicating the supported tiers for enabling staged publishing. At this point, you have the option to select Upgrade and navigate to the Scale tab of your app before continuing.



The first time you add a slot, you only have two choices: clone configuration from the default slot in production or not at all. After you have created several slots, you will be able to clone a configuration from a slot other than the one in production.



✓ There is no content after deployment slot creation. You can deploy to the slot from a different repository branch, or an altogether different repository. You can also change the slot's configuration. Use the publish profile or deployment credentials associated with the deployment slot for content updates.

For more information, you can see:

Add a deployment slot - **https://docs.microsoft.com/en-us/azure/app-service/web-sites-staged-publishing?toc=%2Fazure%2Fapp-service%2Ftoc.json#add-a-deployment-slot**[4]

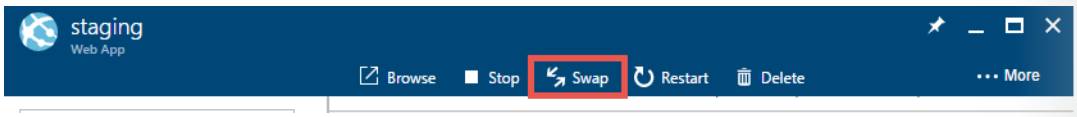
Swap Deployment Slots

When you clone configuration from another deployment slot, the cloned configuration is editable. Furthermore, some configuration elements will follow the content across a swap (not slot specific) while other configuration elements will stay in the same slot after a swap (slot specific).

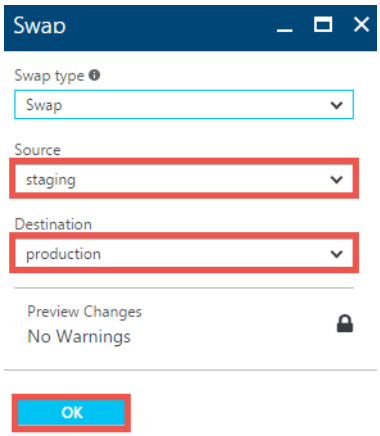
- Settings that **are** swapped: general settings, handler mappings, monitoring, diagnostics, and WebJobs.
- Settings that **are not** swapped: publishing endpoints, custom domain names, SSL certificates and bindings, scale settings, and WebJob schedulers.

4 https://docs.microsoft.com/en-us/azure/app-service/web-sites-staged-publishing?toc=%2Fazure%2Fapp-service%2Ftoc.json

Before you swap an app from a deployment slot into production, make sure that all non-slot specific settings are configured exactly as you want to have it in the swap target. To swap deployment slots, click the Swap button in the command bar of the app or in the command bar of a deployment slot.



Make sure that the swap source and swap target are set properly. Usually, the swap target is the production slot.



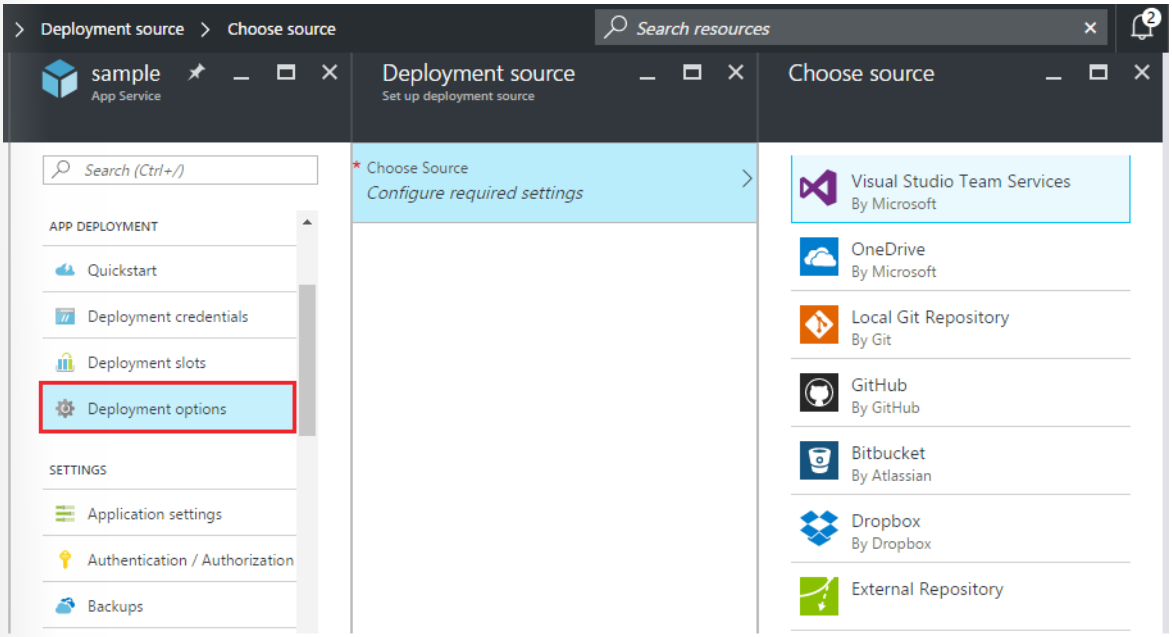
✓ You can configure app settings and connections to stick to a slot and not be swapped. This done in the App Settings blade. A developer can create new settings for the web app. The last video in this lesson reviews this concept.

Demonstration: Deploying to a Stage Slot

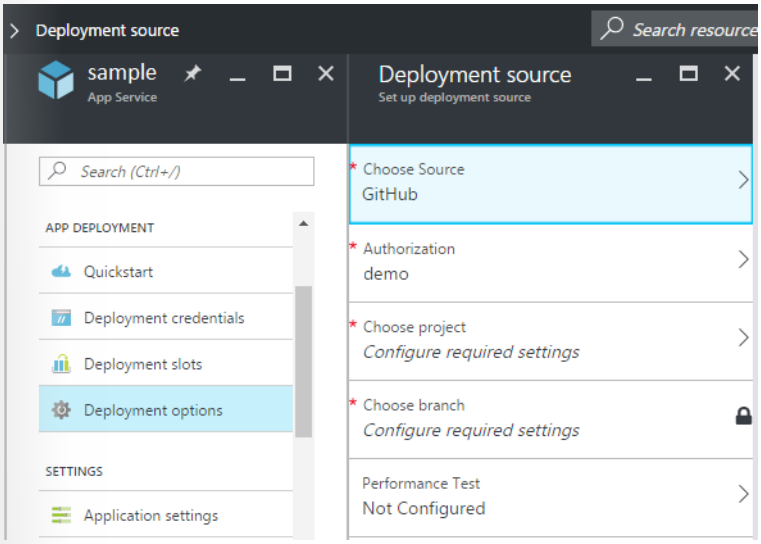
Enabling Continuous Deployment

To enable continuous deployment:

1. Publish your app content to the repository that will be used for continuous deployment.
2. In your app's menu blade in the Azure portal, click **APP DEPLOYMENT > Deployment options**. Click **Choose Source**, then select the deployment source.



1. Complete the authorization workflow.
2. In the **Deployment source** blade, choose the project and branch to deploy from. When you're done, click **OK**.



App Service creates an association with the selected repository, pulls in the files from the specified branch, and maintains a clone of your repository for your App Service app. When you configure VSTS continuous deployment from the Azure portal, the integration uses the App Service Kudu deployment engine, which already automates build and deployment tasks with every git push. You do not need to separately set up continuous deployment in VSTS. After this process completes, the **Deployment options** app blade will show an active deployment that indicates deployment has succeeded.

1. To verify the app is successfully deployed, click the **URL** at the top of the app's blade in the Azure portal.

2. To verify that continuous deployment is occurring from the repository of your choice, push a change to the repository. Your app should update to reflect the changes shortly after the push to the repository completes. You can verify that it has pulled in the update in the **Deployment options** blade of your app.

✓ If you want to set up an alternate repository for your web app, you can easily disable continuous deployment. In your app's menu blade in the Azure portal, click **APP DEPLOYMENT > Deployment options**. Then click **Disconnect** in the **Deployment options** blade.

For more information, you can see:

Continuous Deployment to Azure Web Service - **https://docs.microsoft.com/en-us/azure/app-service/app-service-continuous-deployment**

Demonstration Continuous Deployment

Demonstration: Continuous Deployment

This demonstration shows how to configure a Deployment Source (GitHub) in the Azure Portal. The actual code (beginning at about 7:00) is more targeted for developers, but you may find it out interest. As a System Administrator you will depend on a developer to create the code and give you information about where the source code is located.

Video Managing App Setting and Connection Strings

Video: Managing App Setting and Connection Strings

This is an advanced video that reviews, from a developer point of view, how new web app settings are defined. For the IT Admin this is good general information to understand. The video reviews “sticky” settings and swapping slots. Which of these tasks do you think you will be responsible for?

Practice: Deployment Slots

Take a few minutes to complete the **Add a deployment slot**[5] practice and the **Swap deployment slots**[6] practice. As you have time continue through the other tutorials. In this tutorial, you will learn how to:

- Add a deployment slot.
- Swap deployment slots.
- Configure auto swap.
- Monitor swap progress.
- Delete a deployment slot.

For more information, you can see:

Set up staging environments for web apps - **https://docs.microsoft.com/en-us/azure/app-service/web-sites-staged-publishing**

Web App Templates

Azure Resource Manager has many templates for the Web Apps feature of Azure App Service. The templates can generally be divided into: Deploying a Web App, Configuring a Web App, Linux Web App, Web App with Connected Resources, and App Service Environment for PowerApps.

- **Deploying a Web App**. For example, deploy an Azure web app that pulls code from GitHub, and a Web app with custom deployment slots.
- **Configuring a Web App**. For example, deploy a web app with a custom domain name.
- **Linux Web App**. For example, deploy an Azure web app on Linux with Azure Database for PostgreSQL or Azure Database with MySQL.
- **Web Apps with Connected Resources**. For example, deploy an Azure web app with an Azure Blob storage connection string, and deploy an Azure web app and a SQL database at the Basic service level.
- **App Service Environment for PowerApps**. For example, create an App Service environment v2 in your virtual network.

✓ Defining dependencies for web apps requires an understanding of how the resources within a web app interact. If you specify dependencies in an incorrect order, you might cause deployment errors or create a race condition that stalls the deployment. Read more about this in the reference link.

For more information, you can see:

Azure Resource Manager templates for Web Apps - **https://docs.microsoft.com/en-us/azure/app-service/app-service-rm-template-samples**

5 https://docs.microsoft.com/en-us/azure/app-service/web-sites-staged-publishing
6 https://docs.microsoft.com/en-us/azure/app-service/web-sites-staged-publishing

Guidance on deploying web apps with Azure Resource Manager templates - **Guidance on deploying web apps with Azure Resource Manager templates**[7]

7 https://docs.microsoft.com/en-us/azure/app-service/web-sites-rm-template-guidance

Managing Web Apps

Video Management Tools and Options

Video: Management Tools and Options

Azure provides a variety of ways to manage your web app including: Azure PowerShell, Command Line Interface (CLI), Rest API, and the Azure Portal. This video shows how to navigate the portal to App Service features such as: Settings, App Development, App Service Plans, Support and Troubleshooting, Monitoring, Mobile, and API.

✓ After watching the video, access the Azure Portal and try navigating to the App Service features.

Video Managing App Settings Overview

Video: Managing App Settings Overview

This video shows you how to use the Azure Portal to locate your web apps and discusses many of the general settings that are available. This is a high-level presentation. Additional information about the settings will be covered in the next video.

Video Managing General App Settings

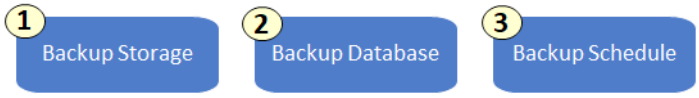
Video: Managing General App Settings

This video continues the coverage of general web app settings. Which of these settings do you think will be most important to your organization?

✓ After viewing the video take a few minutes to access your web app and get familiar with the general settings.

Backup Your App

The Azure App Service backup feature lets you easily create app backups manually or on a schedule. When configuring your backups consider three things: backup storage, backup database, and backup schedule.



1. **Backup storage**. Choose your backup destination by selecting a Storage Account and Container. The storage account must belong to the same subscription as the app you want to backup. It is probably a good idea to create a new storage account or a new container just for backups.
2. **Backup database**. For a database to appear in the backup list, its connection string must exist in the **Connectionstrings** section of the Application settings page for your app. The backup feature supports the following database solutions:

- **SQL Database**[8]
- **Azure Database for MySQL**[9]
- **Azure Database for PostgreSQL**[10]
- **MySQL in-app**[11]

1. **Backup schedule**. Creating a backup schedule is an easy way to automate the backup process. You can configure how often to backup (hours/days), when to backup (calendar with times), and how long to keep the backup (retention days).

By default, your app configuration, file content, and database content is backed up. Each backup is a complete offline copy of your app, not an incremental update. See the reference link on how to configure a partial backup.

✓ Take a few minutes to use the reference link and create a backup of your app.

For more information, you can see:

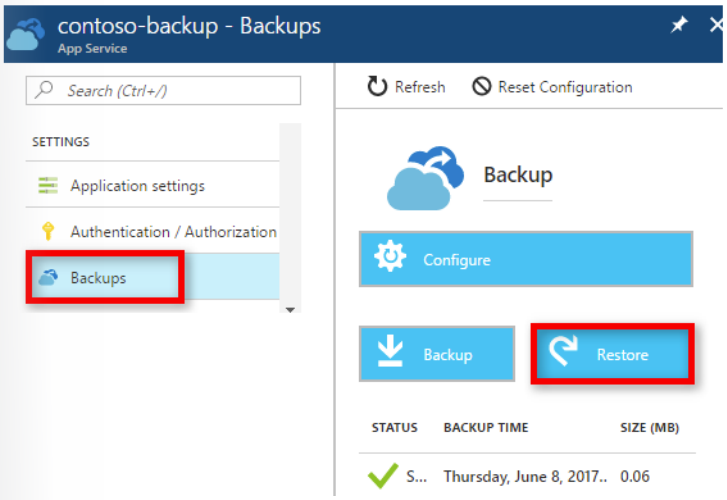
Back up your App in Azure - **https://docs.microsoft.com/en-us/azure/app-service/web-sites-backup**

Configure partial backups - **https://docs.microsoft.com/en-us/azure/app-service/web-sites-backup**

8 https://azure.microsoft.com/services/sql-database/
9 https://azure.microsoft.com/services/mysql
10 https://azure.microsoft.com/en-us/services/postgresql/
11 https://blogs.msdn.microsoft.com/appserviceteam/2017/03/06/announcing-general-availability-for-mysql-in-app

Restore a Backup

After you have backed up your app you can restore the app, with its linked databases, to a previous state, or create a new app using the backup. Notice in the next image, the Restore button and the Backup button discussed in the previous topic.



When configuring a restore there are two things to consider: The **Restore source** and the **Restore destination**.

- **Restore source**. You can select **App Backup** to select any previous existing backup of the current app. You can also select **Storage** and select any backup ZIP file from any existing Azure Storage account and container in your subscription.
- **Restore destination**. You can select **Existing App** to restore the app backup to another app in the same resource group. Before you use this option, you should have already created another app in your resource group with mirroring database configuration to the one defined in the app backup. You can also **Create a New app** to restore your content to.

✓ Restoring from backups is available to apps running in the **Standard** and **Premium** tiers. The **Premium** tier allows a greater number of daily backups to be performed than the **Standard** tier. Take a few minutes to explore and restore an app.

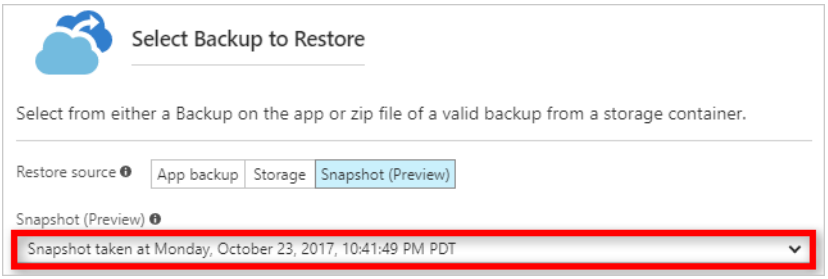
For more information, you can see:

Restore an app in Azure - **https://docs.microsoft.com/en-us/azure/app-service/web-sites-restore**

Restore a Snapshot

If your app is running in the **Premium** tier or higher, the platform automatically saves snapshots for data recovery purposes. Snapshots are incremental shadow copies that have several advantages over backups:

- No file copy errors due to file locks.
- No storage size limitation.
- No configuration required.



At the time of this writing, the snapshot feature is in preview. Also, here are a few other things to know:

- You can only restore to the same app or to a slot belonging to that app.
- App Service stops the target app or target slot while doing the restore.
- App Service keeps three months' worth of snapshots for platform data recovery purposes.
- You can only restore snapshots for the last 30 days.

✓ The Restore destination can either be: **Overwrite** or **New or Existing App**. If you choose **Overwrite**, all existing data in your app's current file system is erased and overwritten. Before you click **OK**, make sure that it is what you want to do.

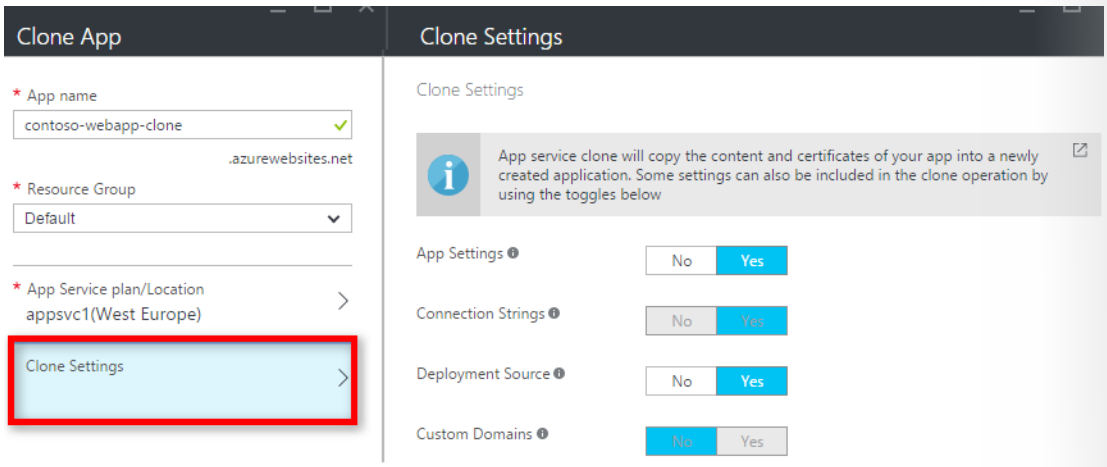
For more information, you can see:

Restore an app in Azure from a snapshot - **https://docs.microsoft.com/en-us/azure/app-service/app-service-web-restore-snapshots**

Cloning an App

The cloning feature in Azure App Service Web Apps lets you easily clone existing web apps to a newly created app in a different region or in the same region. This feature lets you to deploy apps across different regions quickly and easily. App cloning is currently only supported for Premium tier app service plans.

You can clone an app in the Azure Portal or with Azure PowerShell. When you use the portal, you can decide which settings are cloned. For example, you can choose to include App Settings, Connection Strings, Deployment Source, and Custom Domains.



Current Restrictions

As of this time of writing, app cloning is currently in preview, and new capabilities are added over time. There are some restrictions in the current version of app cloning. Here is a list of what is not currently cloned as well as any impacts of app cloning:

- Auto scale settings
- Backup schedule settings
- VNET settings
- Easy Auth settings
- Kudu Extensions
- TiP rules
- Database content
- App Insights are not automatically set up on the destination web app.
- Outbound IP Addresses change if cloning to a different scale unit.

✓ Can you see a need for app cloning in your organization?

For more information , you can see:

Web App Cloning - **https://azure.microsoft.com/en-us/updates/azure-app-service-cloning-available-between-regions/**

Practice: Azure Backup and Restore

The Backup and Restore feature in Azure App Service lets you easily create app backups manually or on a schedule. You can restore the app to a snapshot of a previous state by overwriting the existing app or restoring to another app.

Either create a web app or use an existing one, and take some time to work with the **backup and restore**[12] functionality in the Azure portal.

✓ Be sure to read through the **requirements and restrictions**[13] for backup and restore in the documentation.

In this exercise, you will:

- Create a manual backup of an app.
- Configure an automated backup
- Configure a partial backup

12 https://docs.microsoft.com/en-us/azure/app-service/web-sites-backup
13 https://docs.microsoft.com/en-us/azure/app-service/web-sites-backup

Restoring from backups is available to apps running in **Standard** and **Premium** tier. To work with restore, go to **Restore an app in Azure**[14] and try the following:

- Restore an app from an existing backup
- Download or delete a backup from a storage account
- Monitor a restore operation

You can also try the same tasks using sample scripts, either **PowerShell**[15] or the **Azure CLI**[16].

For more information, you can see:

Back up your App in Azure - **https://docs.microsoft.com/en-us/azure/app-service/web-sites-backup**

Configure partial backups - **https://docs.microsoft.com/en-us/azure/app-service/web-sites-backup**

Restore an app in Azure - **https://docs.microsoft.com/en-us/azure/app-service/web-sites-restore**

14 https://docs.microsoft.com/en-us/azure/app-service/web-sites-restore

15 https://docs.microsoft.com/en-us/azure/app-service/scripts/app-service-powershell-backup-onetime?toc=%2fpowershell%2f-module%2ftoc.json

16 https://docs.microsoft.com/en-us/azure/app-service/scripts/app-service-cli-backup-onetime?toc=%2fcli%2fazure%2ftoc.json

App Service Security

Video: App Security Features Overview

App Service Security Levels

The Azure App Service has two security levels.

- **Infrastructure and platform security**. You trust Azure to provide an infrastructure and platform to securely run your services.
- **Application security**. You design an app with security features. This includes how you integrate with Azure Active Directory, how you manage certificates, and how you make sure that you can securely communicate with different services.

Infrastructure and platform security

The App Service maintains Azure VMs, storage, network connections, web frameworks, management, and integration features, and much more. The App Service is actively secured and hardened and goes through vigorous compliance checks on a continuous basis. These security checks ensure:

- Your App Service apps are isolated from both the Internet and from the other customers' Azure resources.
- Communication of secrets (e.g. connection strings) between your App Service app and other Azure resources (e.g. SQL Database) in a resource group stays within Azure and doesn't cross any network boundaries. Secrets are always encrypted.
- All communication between your App Service app and external resources, such as PowerShell management, command-line interface, Azure SDKs, REST APIs, and hybrid connections, are properly encrypted.
- 24-hour threat management protects App Service resources from malware, distributed denial-of-service (DDoS), man-in-the-middle (MITM), and other threats.

Application security

While Azure is responsible for securing the infrastructure and platform that your application runs on, it is your responsibility to secure your application itself. In other words, you need to develop, deploy, and

manage your application code and content in a secure way. Without this, your application code or content can still be vulnerable to threats such as:

- **SQL Injection**. SQL injection is an attack in which malicious code is inserted into strings that are later passed to an instance of SQL Server for parsing and execution. Any procedure that constructs SQL statements should be reviewed for injection vulnerabilities because SQL Server will execute all syntactically valid queries that it receives. Even parameterized data can be manipulated by a skilled and determined attacker.
- **Session hijacking**. There are two types of session hijacking depending on how they are done. If the attacker directly gets involved with the target, it is called active hijacking, and if an attacker just passively monitors the traffic, it is passive hijacking.
- **Cross-site-scripting**. Cross site scripting attacks work by embedding script tags in URLs and enticing users to click them, ensuring that the malicious script gets executed on the user's computer. These attacks leverage the trust between the user and the server and the fact that there is no input/output validation on the server to reject script language characters. Most browsers are installed with the capability to run scripts enabled by default.
- **Application level Man-In-the-Middle (MITM)**. A MITM attack occurs when an attacker reroutes communication between two users through the attacker's computer without the knowledge of the two communicating users. The attacker can monitor and read the traffic before sending it on to the intended recipient.

For more information , you can see:

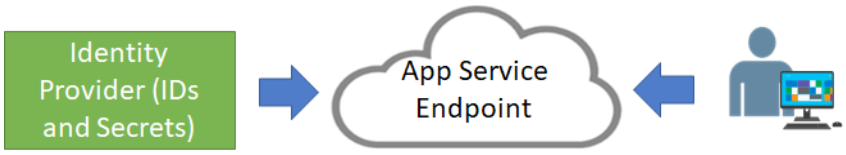
Azure Security Center - **https://azure.microsoft.com/en-us/services/security-center/**

SQL Injection - **https://docs.microsoft.com/en-us/sql/relational-databases/security/sql-injection?view=sql-server-2017**

Session Hijacking - **https://www.greycampus.com/opencampus/ethical-hacking/session-hijacking-and-its-types**

App Service Authentication

How App Service *authentication* works



To authenticate by using one of the identity providers, you first need to configure the identity provider to know about your application. The identity provider will then provide IDs and secrets that you provide to App Service. This completes the trust relationship so that App Service can validate user assertions, such as authentication tokens, from the identity provider.

To sign in a user by using one of these providers, the user must be redirected to an endpoint that signs in users for that provider. If customers are using a web browser, you can have App Service automatically direct all unauthenticated users to the endpoint that signs in users. Otherwise, you will need to direct your customers to *{your App Service base URL}/.auth/login/<provider>*, where <provider> is one of the following values: AAD, Facebook, Google, Microsoft, or Twitter.

Users who interact with your application through a web browser will have a cookie set so that they can remain authenticated as they browse your application. For other client types, such as mobile, a JSON web token (JWT), which should be presented in the X-ZUMO-AUTH header, will be issued to the client. The Mobile Apps client SDKs will handle this for you. Alternatively, an Azure Active Directory identity token or access token may be directly included in the Authorization header as a bearer token.

✓ You're not required to use App Service for authentication and authorization. Many web frameworks are bundled with security features, and you can use them if you like. If you need more flexibility than App Service provides, you can also write your own utilities

For more information, you can see:

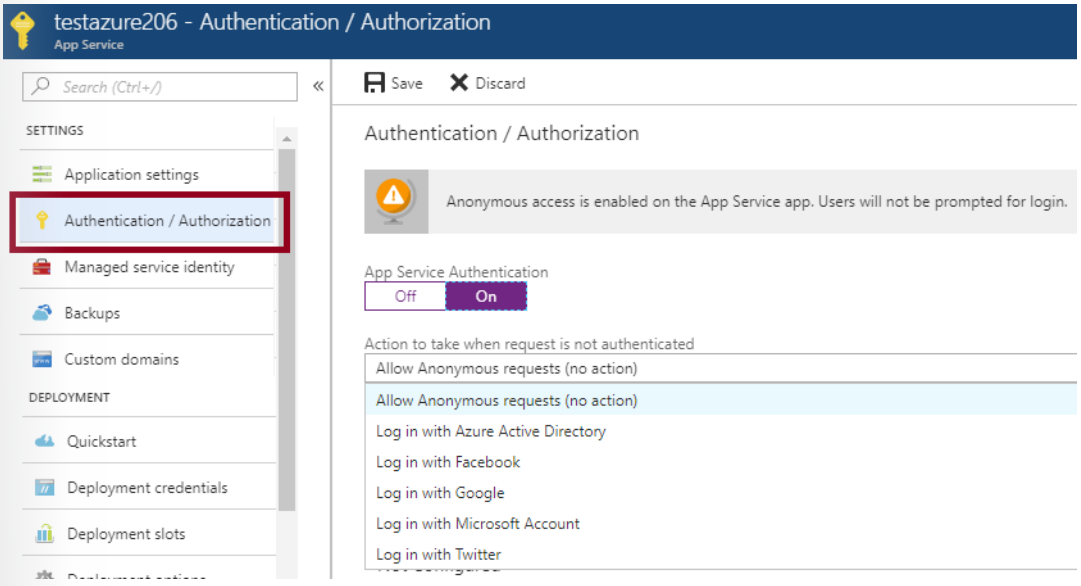
Authentication and authorization in Azure App Service - **https://docs.microsoft.com/en-us/azure/app-service/app-service-authentication-overview**

Authentication Providers

The App Service Authentication / Authorization feature provides a way for your application to sign in users so that you don't have to change code on the app backend. The App Service uses federated identity, in which a third-party identity provider stores accounts and authenticates users. The application relies on the provider's identity information so that the app doesn't have to store that information itself.

The App Service supports five identity providers out of the box: Azure Active Directory, Facebook, Google, Microsoft Account, and Twitter. Your app can use any number of these identity providers to provide your users with options for how they sign in. To expand the built-in support, you can integrate another identity provider or your own custom identity solution.

By default, anonymous access is allowed on the App Service app. To make your App Service more secure you can enable App Service Authentication in the Settings panel.



When App Service authentication is enabled there are several authentication options.

- **Authenticate with Azure Active Directory**[17]. Express mode will use your default AAD and create an AAD application for you. There is also a custom mode if you would like more control over the configuration.

17 https://docs.microsoft.com/en-us/azure/app-service/app-service-mobile-how-to-configure-active-directory-authentication

- **Authenticate with Facebook**[18]. You will need your App ID and App Secret from the **Facebook Developer's website**[19].
- **Authenticate with Google**[20]. You will need the client ID and client secret from the **Google APIs**[21] website.
- **Authenticate with Microsoft Account**[22]. You will need the Application ID and Password values from the **My Applications page**[23] in the Microsoft Account Developer Center.
- **Authenticate with Twitter**[24]. You must have a Twitter account that has a verified email address and phone number. To create a new Twitter account, go to **twitter.com**[25].

✓ Take a minute to create an App Service and view the Authentication and Authorization Settings. Try each of the drop-down options to see what additional information is required. Which option are you most interested in?

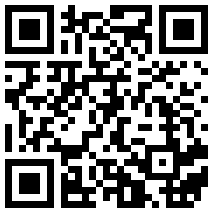
Video: App Service Authentication Options

Demonstration: Authentication and Authorization Features

Demonstration: App Service Security Features

18 https://docs.microsoft.com/en-us/azure/app-service/app-service-mobile-how-to-configure-facebook-authentication
19 http://go.microsoft.com/fwlink/p/?LinkId=268286
20 https://docs.microsoft.com/en-us/azure/app-service/app-service-mobile-how-to-configure-google-authentication
21 http://go.microsoft.com/fwlink/p/?LinkId=268303
22 https://docs.microsoft.com/en-us/azure/app-service/app-service-mobile-how-to-configure-microsoft-authentication
23 http://go.microsoft.com/fwlink/p/?LinkId=262039
24 https://docs.microsoft.com/en-us/azure/app-service/app-service-mobile-how-to-configure-twitter-authentication
25 http://go.microsoft.com/fwlink/p/?LinkID=268287

Video: App Service Isolation Overview

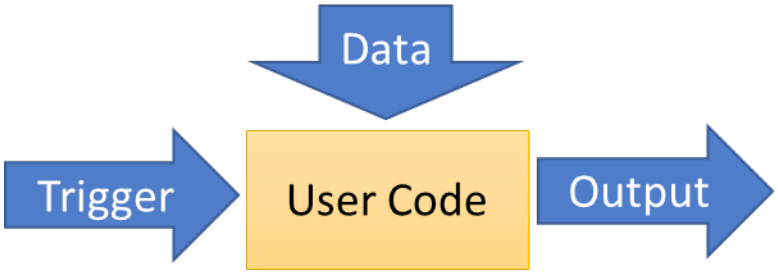


Demonstration: App Service Isolation

Serverless Computing Concepts

Serverless Computing

Developers focus most of their time building and deploying apps, but must often spend time also managing the servers those apps will run on. Having to consider the infrastructure is critical but consumes time that could be spent on app development. Serverless computing provides a real solution to this challenge. Serverless computing is the abstraction of servers, infrastructure, and operating systems. When building serverless apps, developers don't need to provision and manage any servers, and can take their minds off infrastructure concerns.



Infrastructure spun-up, scaled, and spun-down when no longer needed

Serverless computing is driven by the reaction to events and triggers happening in near-real-time—in the cloud. As a fully managed service, server management and capacity planning are invisible to the developer and billing is based just on resources consumed or the actual time your code is running.

Serverless computing has many advantages. Here are a few:

- **Benefit from a fully managed service**. Organizations can relieve their teams from the burden of managing servers. By utilizing fully managed services, developers focus on application business logic and avoid administrative tasks. With serverless architecture developers simply deploy their code, and it runs with high availability.
- **Scale flexibly**. Serverless compute scales from nothing to handling tens of thousands of concurrent functions almost instantly (within seconds), to match any workload, and without requiring scale configuration.
- **Only pay for the resources used**. With serverless architecture, your organization only pays for the time the application code is running. Serverless computing is event-driven, and resources are allocated as soon as they're triggered by an event. You're only charged for the time and resources it takes to execute the application code—through sub-second billing.

For more information, you can see:

Serverless computing - **https://azure.microsoft.com/en-us/overview/serverless-computing/**

Serverless Applications

Serverless computing covers a wide area and has many applications. In this course we will cover four of the main applications in the areas of compute, cloud messaging, and workflow orchestration.

Compute

- **Azure Functions** is an event-driven compute experience that allows an app developer to execute code, written in the programming language of their choice, without worrying about servers. An organization benefits from scale on demand without incurring charges for idle capacity.

Cloud Messaging

- **Event Grid** is a fully managed event routing service that enables rich application scenarios by connecting serverless logic to events coming from multiple Azure services or from a developer or organization's custom apps.
- **Service Bus** is a fully managed messaging infrastructure that enables an organization to build distributed and scalable cloud solutions with connections across private and public cloud environments.

Workflow Orchestration

- **Logic Apps** provide serverless workflows that allow developers to easily integrate data with their apps instead of writing complex glue code between disparate systems. Logic Apps also allow the orchestration and connecting of the serverless functions and APIs in an application.

✓ Processing background jobs, **WebJobs**, will also be covered in this module.

For more information, you can see:

Azure Serverless Computing Cookbook - **https://azure.microsoft.com/en-us/resources/azure-serverless-computing-cookbook/**

Video: Serverless Computing

Comparing Serverless Options

After learning about the different serverless computing options in this module you may be wondering which to choose for your application. Here is a quick summary of what is to come.

Azure Functions vs Logic Apps

Functions and Logic Apps are Azure services that enable serverless workloads. Azure Functions is a serverless compute service, while Azure Logic Apps provides serverless workflows. The following table describes different aspects of Azure Functions and Logic Apps

	Durable Functions	Logic Apps
Development	Code-first (imperative)	Designer-first (declarative)
Connectivity	**About a dozen built-in binding types** *(https://docs.microsoft.com/en-us/azure/azure-functions/functions-triggers-bindings)*, write code for custom bindings	**Large collection of connectors** *(https://docs.microsoft.com/en-us/azure/connectors/apis-list)*, **Enterprise Integration Pack for B2B scenarios** *(https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-enterprise-integration-overview)*, **build custom connectors** *(https://docs.microsoft.com/en-us/azure/logic-apps/custom-connector-overview)*
Actions	Each activity is an Azure function; write code for activity functions	**Large collection of ready-made actions** *(https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-workflow-actions-triggers)*
Monitoring	**Azure Application Insights** *(https://docs.microsoft.com/en-us/azure/application-insights/app-insights-overview)*	**Azure portal** *(https://docs.microsoft.com/en-us/azure/logic-apps/quickstart-create-first-logic-app-workflow)*, **Operations Management Suite** *(https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-monitor-your-logic-apps-oms)*, **Log Analytics** *(https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-monitor-your-logic-apps)*
Management	**REST API** *(https://docs.microsoft.com/en-us/azure/azure-functions/durable-functions-http-api)*, **Visual Studio** *(https://docs.microsoft.com/azure/vs-azure-tools-resources-managing-with-cloud-explorer)*	**Azure portal** *(https://docs.microsoft.com/en-us/azure/logic-apps/quickstart-create-first-logic-app-workflow)*, **REST API** *(https://docs.microsoft.com/rest/api/logic/)*, **PowerShell** *(https://docs.microsoft.com/powershell/module/azurerm.logicapp/?view=azurermps-5.6.0)*, **Visual Studio** *(https://docs.microsoft.com/azure/logic-apps/manage-logic-apps-with-visual-studio)*
Execution context	Can run **locally** *(https://docs.microsoft.com/en-us/azure/azure-functions/functions-runtime-overview)* or in the cloud.	Runs only in the cloud.

Functions and WebJobs

The WebJobs feature of App Service enables you to run a script or code in the context of an App Service web app. Azure Functions is built on the WebJobs SDK, so it shares many of the same event triggers and connections to other Azure services.

	Functions	WebJobs with WebJobs SDK
Serverless app model *(https://azure.microsoft.com/overview/serverless-computing/)* with **automatic scaling** *(https://docs.microsoft.com/en-us/azure/azure-functions/functions-scale)*	Yes	No
Develop and test in browser *(https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-first-azure-function)*	Yes	No
Pay-per-use pricing *(https://docs.microsoft.com/en-us/azure/azure-functions/functions-scale)*	Yes	No
Integration with Logic Apps *(https://docs.microsoft.com/en-us/azure/azure-functions/functions-twitter-email)*	Yes	No

✓ You may want to bookmark this page and return when you've learned a bit more.

For more information, you can see:

Compare Azure Functions and Azure Logic Apps - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-compare-logic-apps-ms-flow-webjobs#compare-azure-functions-and-azure-logic-apps**[26]

Compare Functions and WebJobs - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-compare-logic-apps-ms-flow-webjobs#compare-functions-and-webjobs**[27]

26 https://docs.microsoft.com/en-us/azure/azure-functions/functions-compare-logic-apps-ms-flow-webjobs
27 https://docs.microsoft.com/en-us/azure/azure-functions/functions-compare-logic-apps-ms-flow-webjobs

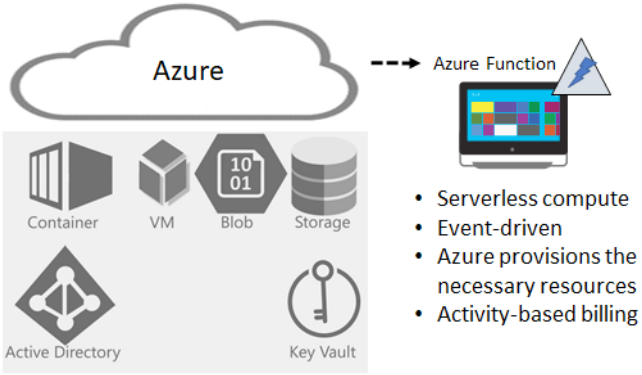
Managing Azure Functions

Video: Azure Functions

Overview of Azure Functions

Azure Functions is a serverless compute service that enables you to run code on-demand without having to explicitly provision or manage infrastructure. Serverless relieves the developer from the operational complexity of running applications. He or she no longer must worry about servers, virtual machines, patching, and scaling. This differs slightly from PaaS, because with PaaS, you still need to choose your operating system and the VM size, which means you need to be able to forecast your demand and then pay for that capacity, even if it's not fully utilized.

With serverless, Azure has compute resources ready to be allocated. Their usage is triggered by an event. The developer provides the code and when an event occurs, such as an Azure alert or when a message is received, Azure provisions the necessary compute resources. This is activity-based billing, so a developer, or the organization, only incurs charges when using the resources.



For more information you can see:

Build apps faster with Azure Serverless - **https://azure.microsoft.com/en-us/blog/build-apps-faster-with-azure-serverless/**

Features of Azure Functions

Azure Functions is a solution for easily running small pieces of code, or "functions," in the cloud. The developer simply writes the code needed for the problem at hand, without worrying about an entire application or the infrastructure to run it.

Features

Here are some key features of Azure Functions that it make it an ideal solution for web app developers:

- **Choice of language**. Write functions using C#, F#, Node.js, Python, PHP, batch, bash, or any executable.
- **Pay-per-use pricing model**. Pay only for the time spent running application code.
- **Bring your own dependencies**. Functions supports NuGet and NPM, allowing use of preferred libraries.
- **Integrated security**. Protect HTTP-triggered functions with OAuth providers such as Azure Active Directory, Facebook, Google, Twitter, and Microsoft Account.
- **Simplified integration**. Easily leverage Azure services and software-as-a-service (SaaS) offerings.
- **Flexible development**. Developers can code their functions directly in the portal or set up continuous integration to deploy their code through GitHub, Visual Studio Team Services, and other supported development tools.
- **Open-source**. The Functions runtime is open-source and available on GitHub.
- **Reuse**. Developers can reuse their functions in multiple applications.

✓ As an administrator you will not be responsible for the coding of the functions, so read this lesson with an eye toward how you will support your app developer.

For more information, you can see:

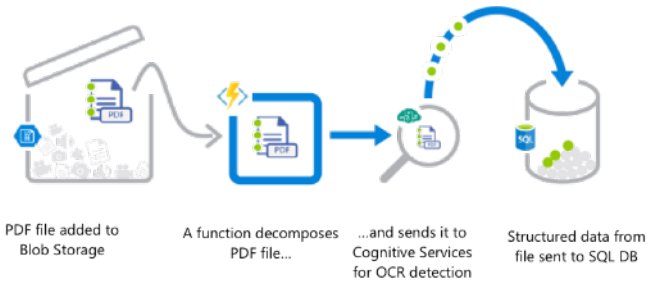
Azure Functions Documentation - **https://docs.microsoft.com/en-us/azure/azure-functions/**

Azure Functions (Examples)

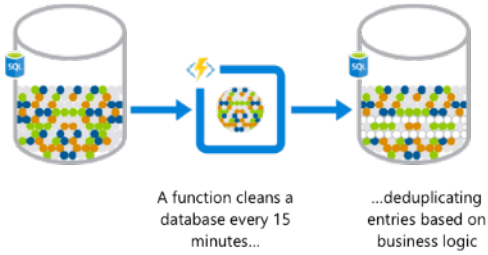
Azure Functions is a great solution for processing data, integrating systems, working with the internet-of-things (IoT), and building simple APIs and microservices. Functions should be considered for tasks like image or order processing, file maintenance, long-running tasks that need to run in a background thread, or for any tasks that run on a schedule. Here are three examples of how you can use Functions.

Web application backends. Online orders are picked up from a queue, processed, with the resulting data stored in a database.

Real-time file processing. Patient records are securely uploaded as PDF files. That data is then decomposed, processed using Optical Character Recognition (OCR) detection, and added to a database for customers who can search the information.



Automation of scheduled tasks. A customer database is analyzed for duplicate entries every 15 minutes. The removal of duplicates ensures multiple communications are not being sent out to same customers.



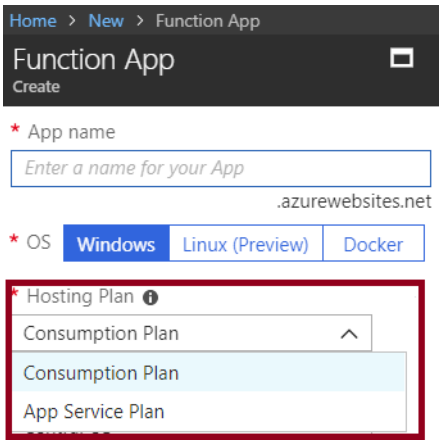
✓ Can you think of any apps that could benefit from functions?

For more information, you can see:

Functions - **https://azure.microsoft.com/en-us/services/functions/**

Function Service Plans

When you create a function app, you must decide on a name, OS, and hosting plan. There are two hosting plans: **Consumption Plan** and **App Service Plan**. Choose the one that best fits your needs.



App Service plan

A developer can run their functions just like any web, mobile, and API apps. While already using App Service for your other applications, the developer can run those functions on the same plan at no additional cost. Currently, Linux hosting is currently only available on an App Service plan.

Consumption plan

When a function runs, Azure provides all the necessary compute resources. The developer or the organization doesn't need to worry about resource management, and only pays for the time that the code runs.

The Consumption plan automatically scales CPU and memory resources by adding additional processing instances based on the runtime needs of the functions in the Function App. The Consumption plan is the default hosting plan and offers the following benefits:

- Pay only when functions are running. Billing is based on number of executions, execution time, and memory used.
- Scale out automatically, even during periods of high load.

✓ Can you see why the Consumption plan is the default?

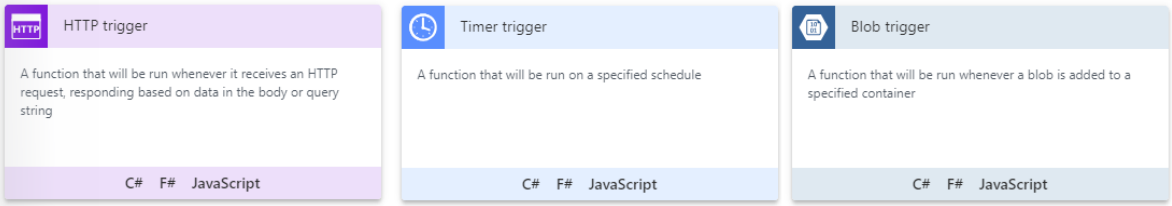
For more information , you can see:

Create an Azure app function - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-scheduled-function#create-an-azure-function-app**[28]

Azure Functions Pricing page - **https://azure.microsoft.com/pricing/details/functions**

Function Templates

After creating a function and selecting the service plan, a developer can use a template for many different key scenarios. The template will create a function triggered to different events. The trigger will start the execution of the code. Here are some example templates.



HTTP Trigger. A function that will run whenever it receives an HTTP request. The function responds based on data in the body or query string.

Timer Trigger. A function that runs on a specified schedule. For example, cleanup or batch tasks.

Blob Trigger. A function that will run whenever a blob is added to a specified container. This function might be used for resizing images that will be added to web pages.

There are other triggers that respond to Event Hub, GitHub, webhook, and queue events. By default, a function will timeout after 5 minutes, and a function can run for a maximum of 10 minutes.

✓ Use the reference link to learn more about triggers.

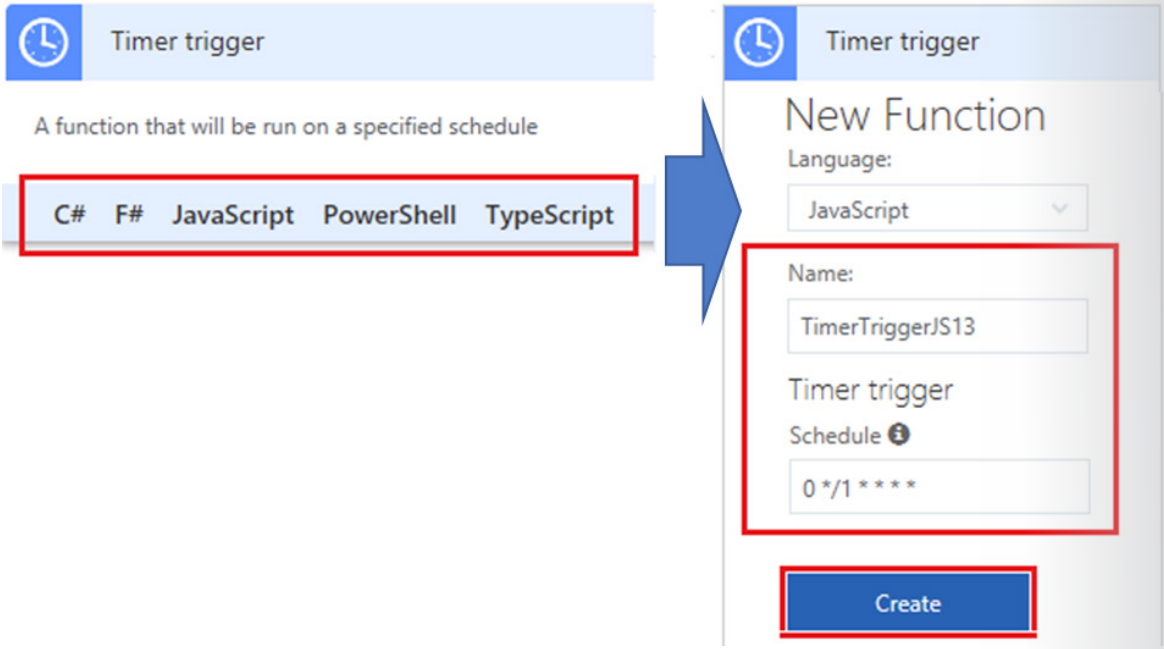
For more information, you can see:

Azure Functions triggers and bindings developer reference - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-triggers-bindings**

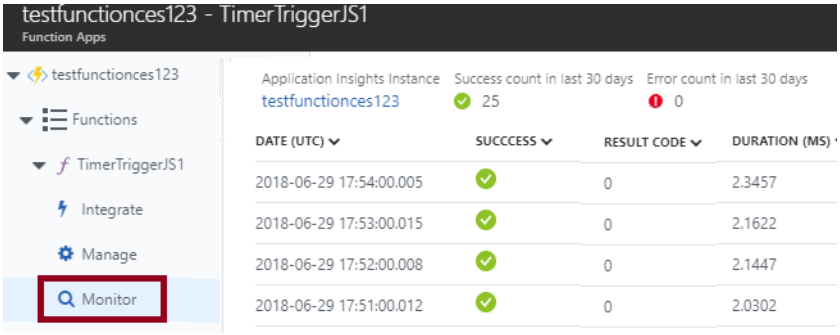
28 https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-scheduled-function

Implementing Functions

Once of the easiest functions to understand is the Timer. After choosing the language, the Timer template only requires the timer name and the schedule. The schedule field is a **CRON expression**[29]. For example, 0 */1 * * * *, means the function will run every minute.



Once your function is running you can monitor its progress.



✓ Use the Manage link to disable or delete your function. Click the name of the function to see the code behind the function. Creating a timer function is one of the practices for this lesson.

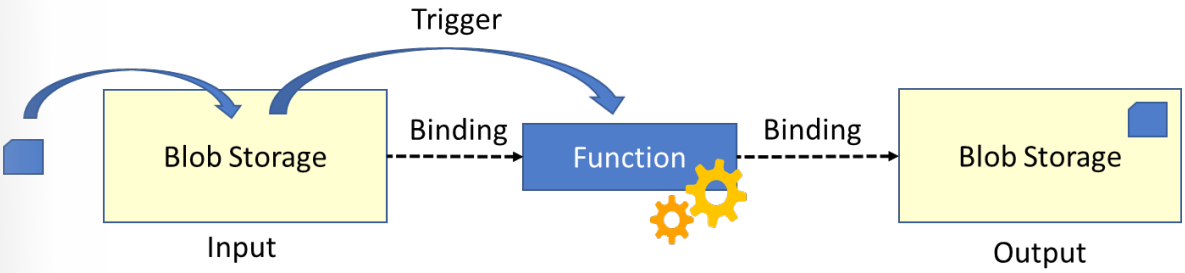
For more information, you can see:

Create a function in Azure that is triggered by a timer - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-scheduled-function#create-an-azure-function-app**[30]

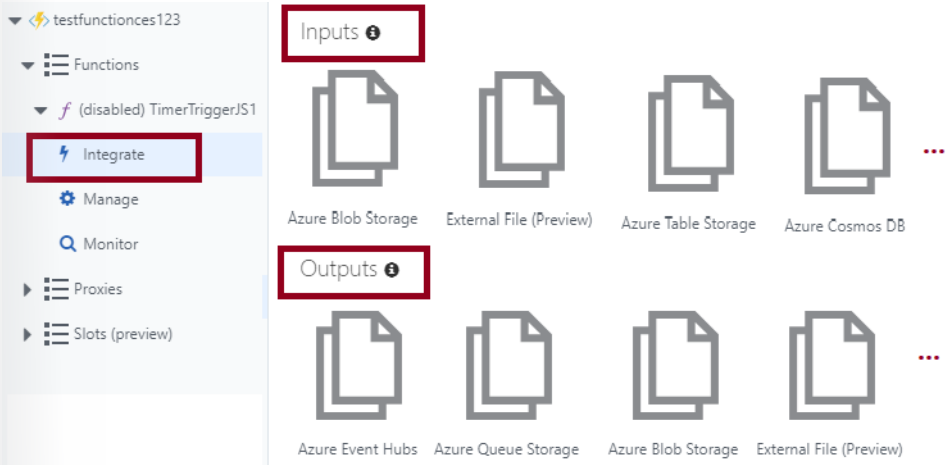
29 http://en.wikipedia.org/wiki/Cron
30 https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-scheduled-function

Bindings

So far you have learned that a trigger defines how a function is invoked. Bindings are a way to provide input and output to the function. For example, if the function is resizing an image then it could be bound to the incoming Blob storage. When an image arrives in that storage, a trigger would start the resizing function. The processed image could then be stored in the same or different Blob storage. A function can have multiple input and output bindings. Bindings are always optional.



To see what bindings are available to your function use the **Integrate** link.



Each input or output binding will require different configuration parameters. For example, if you select Blob storage you would be prompted for the storage account name and the blob storage path.

✓ There are a variety of bindings to choose from and each function will have different bindings that are available. Use the reference link to learn more. Do you have an idea of a function that might need bindings?

For more information, you can see:

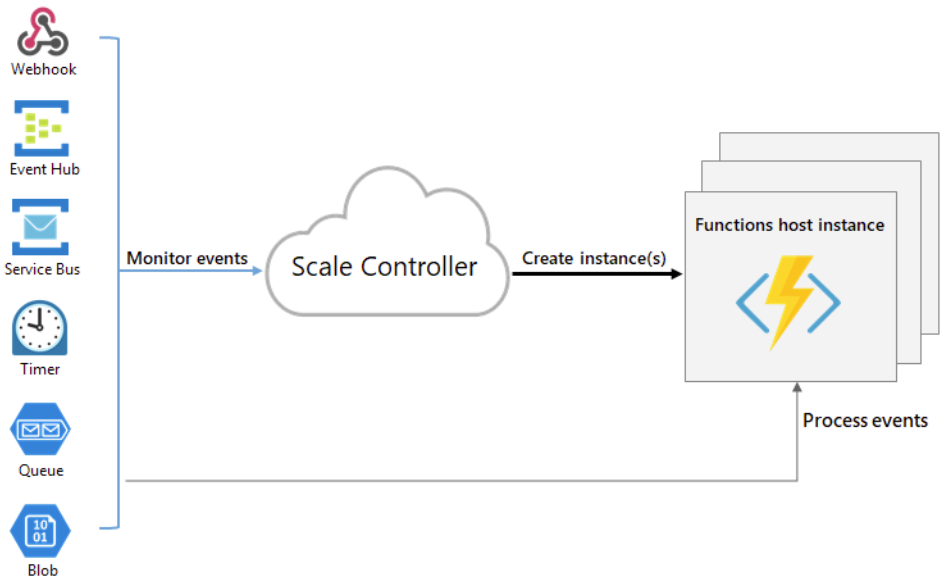
Supported bindings - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-triggers-bindings#supported-bindings**[31]

Function Scaling

Azure Functions can scale to meet your needs. Azure Functions uses a component called the *scale controller* to monitor the rate of events and determine whether to scale out (add host instances) or scale in (remove host instances).

31 https://docs.microsoft.com/en-us/azure/azure-functions/functions-triggers-bindings

The scale controller uses heuristics for each trigger type. For example, when you're using an Azure Queue storage trigger, it scales based on the queue length and the age of the oldest queue message.



The unit of scale is the function app. When the function app is scaled out, additional resources are allocated to run multiple instances of the Azure Functions host. Conversely, as compute demand is reduced, the scale controller removes function host instances. The number of instances is eventually scaled down to zero when no functions are running within a function app.

✓ A single function app will only scale to a maximum of 200 instances. A single instance may process more than one message or request at a time though, so there isn't a set limit on number of concurrent executions. New instances will only be allocated at most once every 10 seconds.

For more information, you can see:

Runtime scaling - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-scale#runtime-scaling**[32]

Scalability best practices - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-best-practices#scalability-best-practices**[33]

Demonstration: Azure Functions

32 https://docs.microsoft.com/en-us/azure/azure-functions/functions-scale
33 https://docs.microsoft.com/en-us/azure/azure-functions/functions-best-practices

Practice: Blob Storage Function

Take a few minutes to **Create a function triggered by Azure Blob storage**[34]. In this exercise you will earn how to create a function triggered when files are uploaded to or updated in Azure Blob storage. In this practice you will learn how to:

- Create an Azure Function app.
- Create a Blob storage triggered function.
- Create the container.
- Test the function.
- Clean up resources.

For more information, you can see:

Azure Functions Blob storage bindings - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-storage-blob**

Practice: Timer Function

Take a few minutes to **Create a function in Azure that is triggered by a timer**[35]. In this exercise you will earn how to create a function that runs based a schedule that you define. In this practice you will learn how to:

- Create an Azure Function app.
- Create a timer triggered function.
- Update the timer schedule.

✓ The Azure documentation has other function examples. As you have time, try the queue storage example at the reference link.

For more information, you can see:

Create a function triggered by Azure Queue storage - **https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-storage-queue-triggered-function**

34 https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-storage-blob-triggered-function
35 https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-scheduled-function

Managing Event Grid

Video: Event Grid Overview

Video: Implementing Event Grid

Overview of Event Grid

Simplify event-based apps with Event Grid, a single service for managing routing of all events from any source to any destination. Designed for high availability, consistent performance, and dynamic scale, Event Grid lets a developer focus on the app logic rather than infrastructure. Here are a few advantages:

Simplify event delivery

Eliminate polling—and the associated cost and latency. With Event Grid, event publishers are decoupled from event subscribers using a pub/sub model and simple HTTP-based event delivery, allowing developers to build scalable serverless applications, microservices, and distributed systems.

Build reliable cloud applications

Gain massive scale, dynamically, while getting near-real-time notifications for changes that the developer or organization is interested in. Build better, more reliable applications through reactive programming, capitalizing on guaranteed event delivery and the high availability of the cloud.

Focus on product innovation

Develop richer application scenarios by connecting multiple possible sources and destinations of events. Your business logic can be triggered by virtually all Azure services, as well as custom sources. Fully managed event delivery, intelligent filtering, and the ability to send events to multiple recipients at once allowing the developer to focus on solving business problems rather than infrastructure.

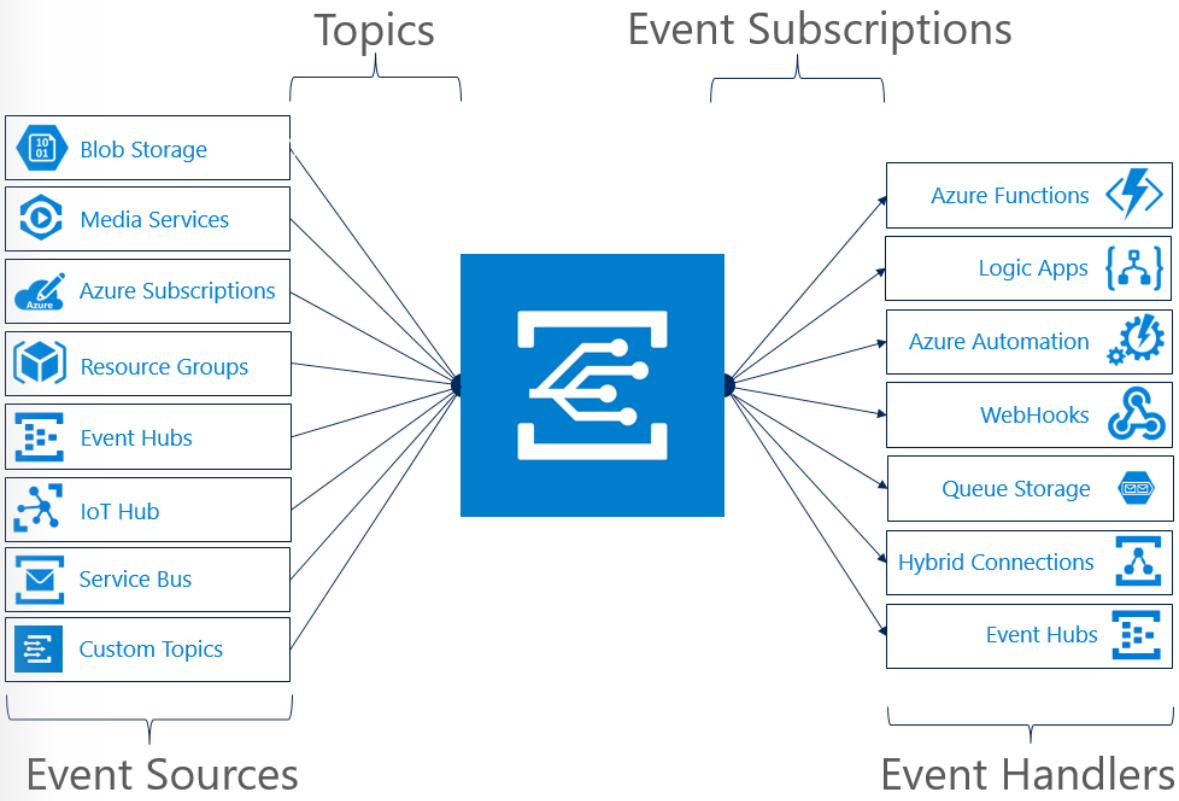
For more information, you can see:

Event Grid - **https://azure.microsoft.com/en-us/services/event-grid/**

Event Grid pricing - **https://azure.microsoft.com/en-us/pricing/details/event-grid/**

Event Grid Concepts

This diagram shows the four basic concepts in Event Grid: Event Source, Topics, Event Subscriptions, and Event Handlers.



Event Source. An event source is where the event happens. Several Azure services are automatically configured to send events. For example, Azure Storage is the event source for blob created events. Developers can also create custom applications that send events. Custom applications do not need to be hosted in Azure to use Event Grid for event distribution.

Topic. The event grid topic provides an endpoint where the source sends events. A topic is used for a collection of related events.

Event Subscription. A subscription tells Event Grid which events on a topic you are interested in receiving. When creating the subscription, you provide an endpoint for handling the event. You can filter the events that are sent to the endpoint.

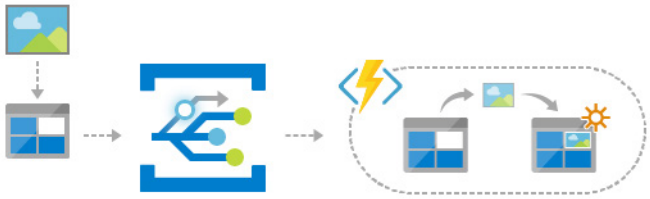
Event Handler. An event handler is the place where the event is sent. The handler takes some further action to process the event. Event Grid supports multiple handler types. For example, Azure Automation, Queue Storage, and Logic Apps.

For more information, you can see:

Event Grid Concepts - **https://docs.microsoft.com/en-us/azure/event-grid/concepts**

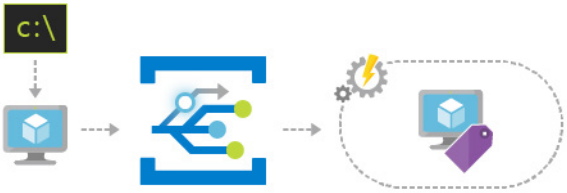
Event Grid Examples

Serverless application architectures



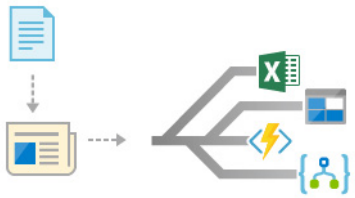
Use Event Grid to instantly trigger a serverless function to run image analysis each time a new photo is added to a blob storage container.

Ops automation



Notify Azure Automation when a virtual machine is created, or a SQL Database is spun up. These events can be used to automatically check that service configurations are compliant, put metadata into operations tools, tag virtual machines, or file work items.

Application integration

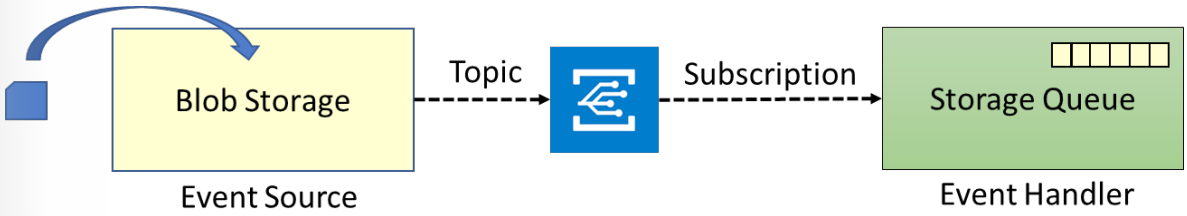


Use Event Grid with serverless computing to process data anywhere, without writing code.

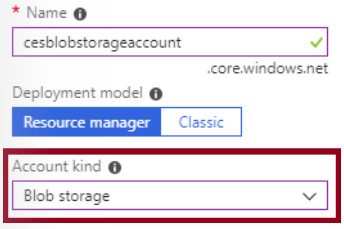
✓ Can you think of any other applications where you can use Event Grid?

Implementing Event Grid (Part 1)

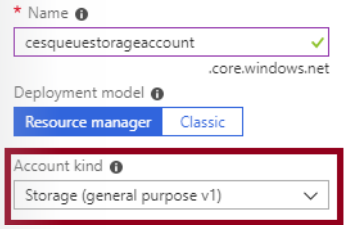
To demonstrate Event Grid let's pick an Event Source (Azure Blob Storage) and an Event Handler (Azure Queue Storage). Let's step through writing a message to a queue when a blob is uploaded.



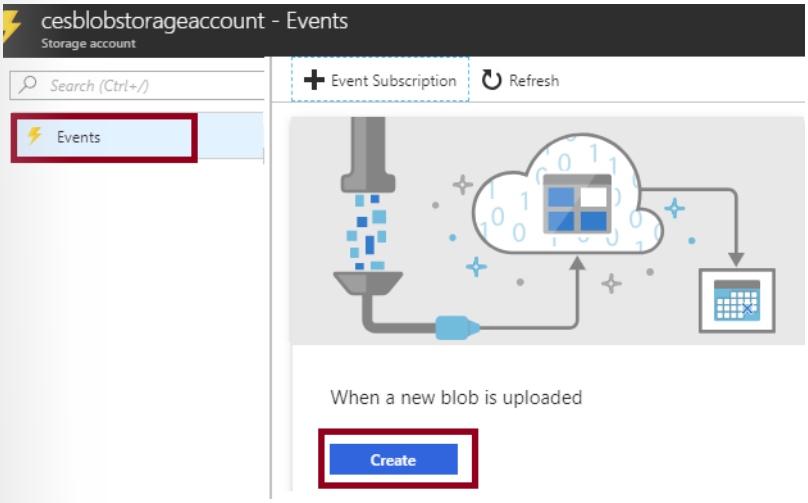
1. Create an Azure Storage account. Ensure the Account Kind is Blob Storage. This is the event source.



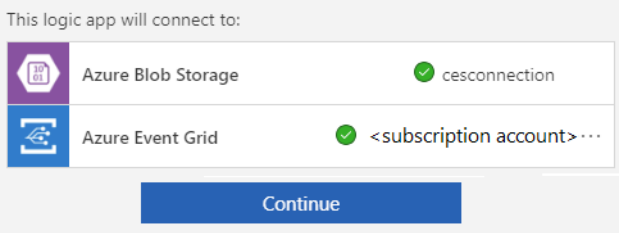
1. Create an Azure Storage Account. Ensure the Account Kind is Standard (general purpose v1). This will be the event handler.



1. Create a queue in the Standard storage account. This will receive the message traffic when a blob is uploaded.
2. Using the Blob storage account and the Events blade, create the When a new blob is updated template. Notice the other templates that are available.



1. When prompted give your Blob storage account a connection name. Also, connect to the Event Grid by providing your subscription information. This will be used to connect to the queue storage.

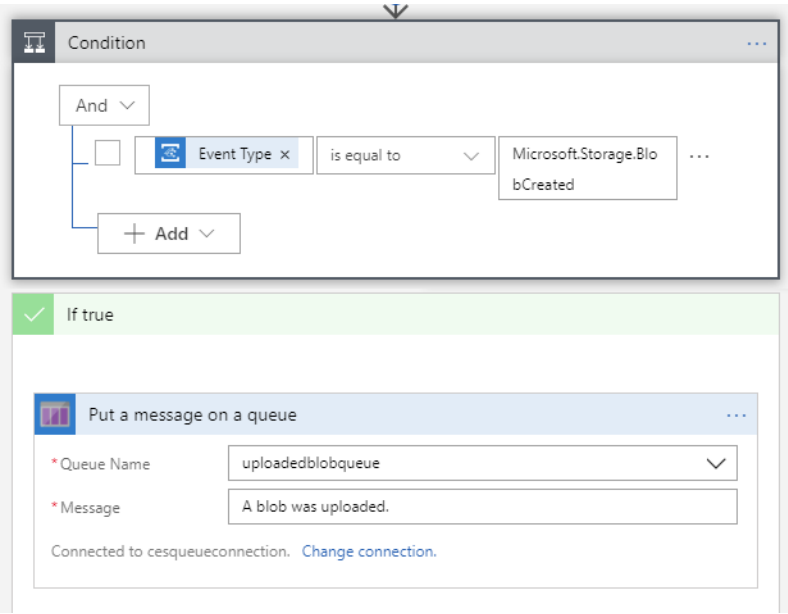


✓ This topic continues on the next page.

Implementing Event Grid (Part 2)

✓ This topic is a continuation of the previous.

1. Edit the Logic App so that it has one condition and one action. The condition is true whenever a blob is created. The action is to put a message in the queue. The original template has several actions that you need to remove. Delete them from the bottom up to avoid dependency problems. Your finished workflow should look like this. For this simple example, the if false action is not specified.



1. Save your changes and run your logic app.
2. Return to the Blob storage account, create a container, and upload a blob.
3. View your Queue storage account and ensure messages have been added to the queue.

ID	MESSAGE TEXT	INSERTION TIME	EXPIRATION TIME	DEQUEUE COUNT
5d9fe07a-efaa-47f1-a587...	A blob was uploaded.	Thu, 05 Jul 2018 23:55:36 GMT	Thu, 12 Jul 2018 23:55:36 GMT	0
8baedddd-44b2-4dc2-a1...	A blob was uploaded.	Thu, 05 Jul 2018 23:55:54 GMT	Thu, 12 Jul 2018 23:55:54 GMT	0

1. To avoid billing charges, remove your storage accounts and logic app.

✓ Can you see the benefits of Event Grid in providing you a way to connect two services or applications without any coding?

Practice: Event Grid

The documentation has many Event Grid Quickstarts and tutorials for the portal, PowerShell, and the CLI. Here are just a few.

- **Create and route Blob storage events with the Azure portal and Event Grid**[36].
- **Route Blob storage events to a custom web endpoint with PowerShell**[37].
- **Monitor virtual machine changes with Azure Event Grid and Logic Apps**[38].

✓ If you don't see something you would like to work on search the documentation for other examples. Perhaps you would like to try one of the Resource Manager templates instead? The reference link will take you there.

For more information, you can see:

Azure Resource Manager templates for Event Grid - **https://docs.microsoft.com/en-us/azure/event-grid/template-samples**

36 https://docs.microsoft.com/en-us/azure/event-grid/blob-event-quickstart-portal
37 https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-event-quickstart-powershell?toc=%2fazure%2fevent-grid%2ftoc.json
38 https://docs.microsoft.com/en-us/azure/event-grid/monitor-virtual-machine-changes-event-grid-logic-app

Managing Service Bus

Video Service Bus

Video: Service Bus

Channel 9 has a series on Azure Bus. This the first (101) part of the series. If you find it interesting take time to view the other videos. Although, this is an older video in the series, and Python code is used in this video we will use the portal in the practice exercise to configure the service bus. Despite the Azure portal interface having changed since, the concepts around service bus are still valid.

Queues

Azure Service Bus is a multi-tenant cloud messaging service that sends information between applications and services. The information is stored in a message queue. The asynchronous operations give you flexible, brokered messaging, along with structured processing, and publish/subscribe capabilities.

Queues offer *First In, First Out* (FIFO) message delivery to one or more competing consumers. That is, receivers typically receive and process messages in the order in which they were added to the queue, and only one message consumer receives and processes each message.

A key benefit of using queues is to achieve "temporal decoupling" of application components. In other words, the producers (senders) and consumers (receivers) do not have to be sending and receiving messages at the same time, because messages are stored durably in the queue. Furthermore, the producer does not have to wait for a reply from the consumer to continue to process and send messages. And, the consumer doesn't have to be online.

For more information, you can see:

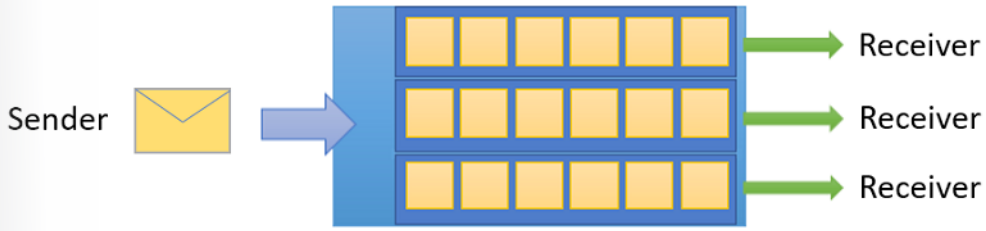
Service Bus - **https://azure.microsoft.com/en-us/services/service-bus/**

Azure Service Bus Messaging Documentation - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/**

Queues - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-queues-topics-subscriptions#queues**[39]

Topics and Subscriptions

In contrast to queues, in which each message is processed by a single consumer, topics and subscriptions provide a one-to-many form of communication, in a publish/subscribe pattern. Useful for scaling to large numbers of recipients, each published message is made available to each subscription registered with the topic.



Topic with three Subscriptions with Messages

Messages are sent to a topic and delivered to one or more associated subscriptions, depending on filter rules that can be set on a per-subscription basis. The subscriptions can use additional filters to restrict the messages that they want to receive. For example, if you were processing delivery orders, rules could be used to identify the nearest subscriber to make the delivery.

Messages are sent to a topic in the same way they are sent to a queue, but messages are not received from the topic directly. Instead, they are received from subscriptions. A topic subscription resembles a virtual queue that receives copies of the messages that are sent to the topic. Messages are received from a subscription identically to the way they are received from a queue.

✓ You can use rules and filters to define conditions that trigger optional **actions**[40], filter specified messages, and set or modify message properties. Do you see the difference between queues and topics?

For more information, you can see:

Topics and subscriptions - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-queues-topics-subscriptions#topics-and-subscriptions**[41]

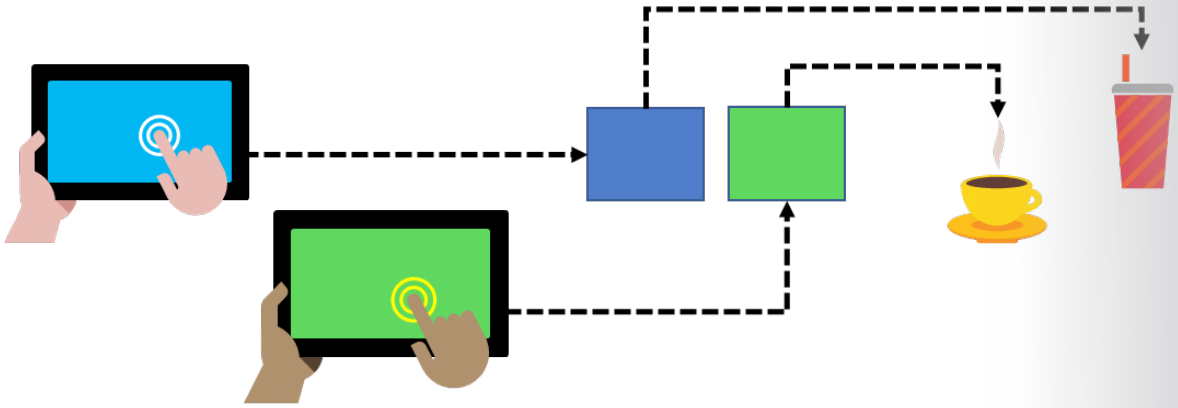
Topic filters and actions - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/topic-filters**

Service bus - **https://azure.microsoft.com/en-us/services/service-bus/**

Service Bus Features

Let's look at the benefits of Service Bus by examining a scenario. In this scenario, you have a web application that takes online orders and several workers ready to fulfill those orders.

39 https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-queues-topics-subscriptions
40 https://docs.microsoft.com/en-us/azure/service-bus-messaging/topic-filters
41 https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-queues-topics-subscriptions



Here are several benefits of using a message queueing system in this situation.

Load Leveling. During the day certain times will generate more orders than at other times. By using a queue, the orders can be stored awaiting fulfillment. The queue automatically increases and decreases in length. This ultimately saves you money because you don't have to pay for a system that is underutilized part of the time. Also, customers do not have to wait to place their order.

Loose Coupling. Your message queues are durable and will reach the worker even if they are busy with another order. This provides a lot of resilience for your ordering system. Consider a scenario where someone else is handling the orders and their system is down. When the system comes back up the orders will still be there.

Load Balancing. When you have more than one fulfillment worker they may work at different speeds. You don't have to programmatically design some way to balance the load, that will naturally occur. You can bring on more workers as the queue increases.

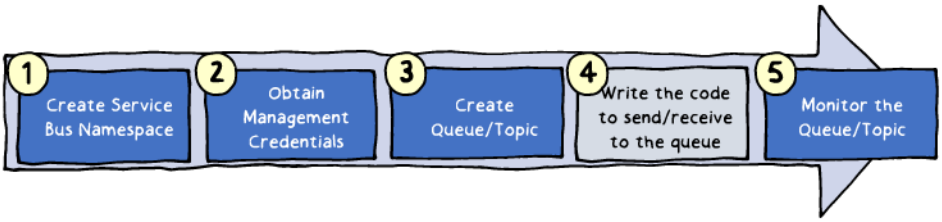
✓ Service bus has many advanced queueing features such as auto-forwarding, batching, scheduled delivery, and message deferral. Read more at the reference link. Can you see how this scenario could be expanded to the topic and subscriber situation?

For more information, you can see:

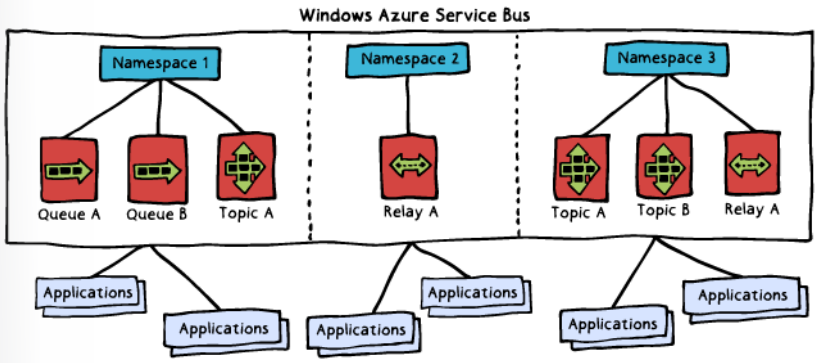
What is Azure Service Bus - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-messaging-overview**

Implementing Service Bus

As a System Administrator you will not be responsible for developing the code to produce or retrieve the messages. You will be responsible for creating the Service Bus namespace, obtaining management credentials, creating the queue/topic, and then managing and monitoring the queues/topics. These steps can be done programmatically but let's look at how to do it in the portal.



First, Service Bus services are typically partitioned into namespaces. Each namespace provides both a service and security boundary. A namespace is a scoping container for all messaging components. Multiple entities can reside within a single namespace, and namespaces often serve as application container.



In the diagram, a Service Bus relay is shown. The Azure Relay service facilitates your hybrid applications by helping you more securely expose services that reside within a corporate enterprise network to the public cloud. You can expose the services without opening a firewall connection, and without requiring intrusive changes to a corporate network infrastructure.

✓ Can you see why you would use multiple namespaces. Will you need to use more than one name-space?

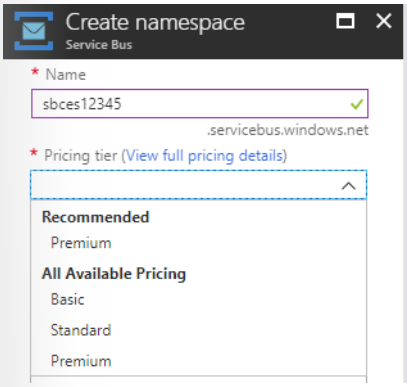
For more information, you can see:

Azure Relay Bus - **https://docs.microsoft.com/en-us/azure/service-bus-relay/relay-what-is-it**

Creating the Namespace

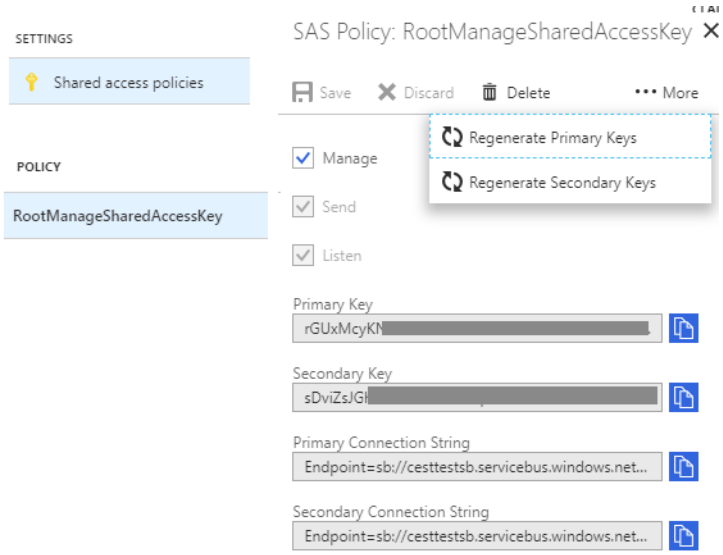
Creating the namespace is easy. The name provides a unique identifier for the object. For example, sbces12345.servicebus.windows.net. Applications can provide this name to Service Bus, then use any entity in the namespace.

Azure Service Bus is offered in Basic, Standard, and **Premium**[42] pricing tiers. You can choose a service tier for each Service Bus service namespace that you create, and this tier selection applies across all entities created within that namespace. Queues are available in all pricing tiers. Topics require a Standard or Premium pricing tier.



42 https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-premium-messaging

Creating a new namespace automatically generates an initial Shared Access Signature (SAS) rule with an associated pair of primary and secondary keys that each grant full control over all aspects of the namespace. Your developer will need the namespace and the connection string.



✓ Be sure to read about the different pricing tiers. Which one do you think you will need? Have you thought about how often you will rotate the SAS keys?

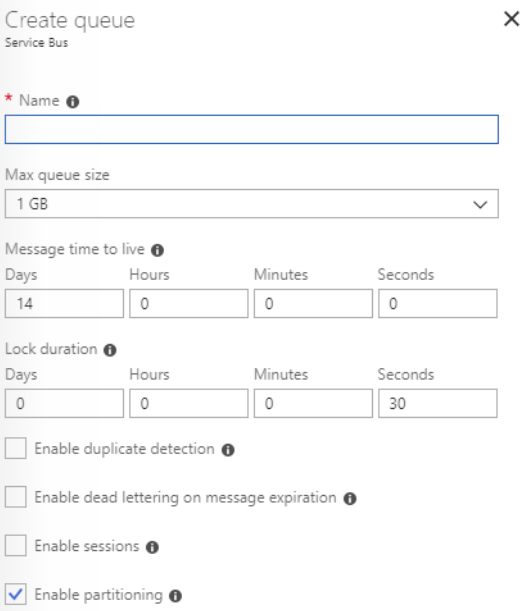
For more information, you can see:

Azure Service Bus Pricing - **https://azure.microsoft.com/en-us/pricing/details/service-bus/**

Service Bus Premium and Standard messaging tiers - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-premium-messaging**

Create a Queue

When you create a queue, you must provide a Name and Max queue size. Additionally, there are some parameters that will affect how the queue performs.



Message time to live. Determines how long a message will stay in the queue before it expires and is removed or dead lettered. This default will be used for all messages in the queue which do not specify a time to live for themselves.

Lock duration. Sets the amount of time a message is locked from other receivers. After its lock expires, a message is pulled by one receiver before being available to be pulled by other receivers. The default is 30 seconds, with a maximum of 5 minutes.

Enable duplicate detection. Configures your queue to keep a history of all messages sent to the queue during a configurable amount of time. During that interval, your queue will not accept any duplicate messages.

Enable dead lettering. Enables holding messages that cannot be successfully delivered to any receiver. The messages are held in a separate queue after they expire. You can inspect this queue.

Enable sessions. Allows ordered handling of unbound sequences of related messages. This guarantees first-in-first-out delivery of messages.

Enable partitioning. Partitions a queue across multiple message brokers and message stores. Partitioning means that the overall throughput of a partitioned entity is no longer limited by the performance of a single message broker or messaging store. In addition, a temporary outage of a messaging store does not render a partitioned queue or topic unavailable.

For more information, you can see:

Message expiration (time to live) - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/message-expiration**

Duplicate detection - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/duplicate-detection**

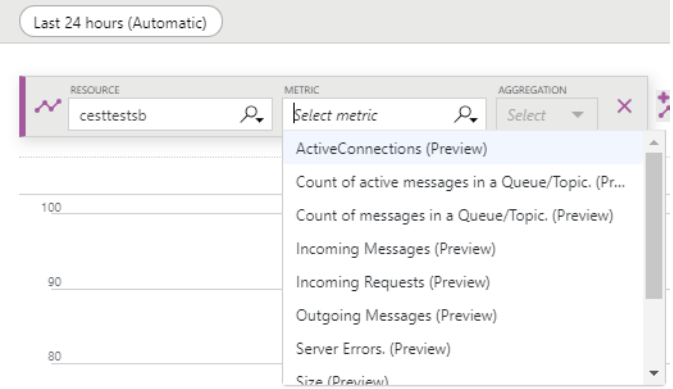
Service Bus dead letter queues -

Message sessions: first in, first out (FIFO) - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/message-sessions**

Partitioned queues and topics - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-partitioning** [43]

Monitoring Service Bus

Service Bus metrics gives you the state of resources in your subscription. With a rich set of metrics data, you can assess the overall health of your Service Bus resources, not only at the namespace level, but also at the queue/topic/message level. These statistics can be important as they help you to monitor the state of Service Bus and help you troubleshoot root-cause issues.



Diagnostic logs

You can configure diagnostic logs for richer information about everything that happens within a job. Diagnostic logs cover activities from the time the job is created until the job is deleted, including updates and activities that occur while the job is running.

For more information, you can see:

Azure Service Bus metrics in Azure Monitor (preview) - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-metrics-azure-monitor**

Service Bus diagnostic logs - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-diagnostic-logs**

Comparing Service Bus and Storage Queues

Azure supports two types of queue mechanisms: Storage queues and Service Bus queues. Now that you have learned about Service Bus queues you may be wondering how they are different from Storage queues. This table provides a summary.

Comparison Criteria	Storage Queues	Service Bus Queues
Ordering guarantee	No	Yes – FIFO
Delivery guarantee	At-Least-Once	At-Least-Once At-Most-Once
Lease/lock level	Message level	Queue level
Batch receive	Yes	Yes
Batch send	No	Yes

43 https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-partitioning

Comparison Criteria	Storage Queues	Service Bus Queues
Scheduled delivery	Yes	Yes
Automatic dead lettering	No	Yes
Message auto-forwarding	No	Yes
Message groups	No	Yes
Duplicate detection	No	Yes

This is a very short table. The reference link provides other information such as queue sizes, message sizes, and authentication options. Notice if your queue size will exceed 80 GB, then you must use storage queues.

✓ By gaining a deeper understanding of the two technologies, you will be able to make a more informed decision on which queue technology to use, and when. Your decision will depend heavily on the individual needs of your application and its architecture. Which option do you think you will use?

For more information, you can see:

Storage queues and Service Bus queues - compared and contrasted - **https://azure.microsoft.com/en-us/documentation/articles/service-bus-azure-and-service-bus-queues-compared-contrasted/**

Practice: Service Bus Message Queues

Take a few minutes to try the **Quickstart: Send and receive messages using the Azure portal and .NET**[44]. In this practice you will earn how:

- Create a Service Bus namespace.
- Obtain management credentials.
- Create a queue.
- Send and receive messages.

✓ Rather than complete the coding part of the practice try to see if you can identify where your connection string and queue information will be used. Also, read the reference link scenario to see how topics can be used in multiple subscription and multiple receiver scenarios.

For more information, you can see:

Tutorial: Update inventory using Azure portal and topics/subscriptions - **https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-tutorial-topics-subscriptions-portal**

Practice: Service Bus Templates

44 https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-quickstart-portal

There are several Azure Resource Manager templates you can use to create namespaces, queues, and topics.

- **Create a namespace**[45]
- **Create a Service Bus namespace with queue**[46]
- **Create a Service Bus namespace with topic and subscription**[47]

✓ To check for the latest templates, visit the **Azure Quickstart Templates**[48] gallery and search for Service Bus.

For more information, you can see:

Azure Quickstart templates - **https://github.com/Azure/azure-quickstart-templates**

45 https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-resource-manager-namespace
46 https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-resource-manager-namespace-queue
47 https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-resource-manager-namespace-topic
48 https://azure.microsoft.com/documentation/templates/?term=service+bus

Managing Logic App

Video: Azure Logic Apps

Video: Getting Started with Logic Apps

Logic Apps

Logic Apps provide a way to simplify and implement scalable integrations and workflows in the cloud. It provides a visual designer to model and automate your process as a series of steps known as a workflow. There are many connectors across the cloud and on-premises to quickly integrate across services and protocols. A logic app begins with a trigger, like 'When an account is added to Dynamics CRM', and after firing, can begin many combinations actions, conversions, and conditional logic.

The advantages of using Logic Apps include the following:

- Getting started quickly from templates.
- Saving time by designing complex processes using easy to understand design tools.
- Implementing patterns and workflows seamlessly, that would otherwise be difficult to implement in code.
- Customizing your logic app with your own custom APIs, code, and actions.
- Connecting and synchronizing disparate systems across on-premises and the cloud.

Logic Apps is a fully managed iPaaS (integration Platform as a Service) freeing users from worry about building hosting, scalability, availability, and management. Logic Apps will scale up automatically to meet demand.

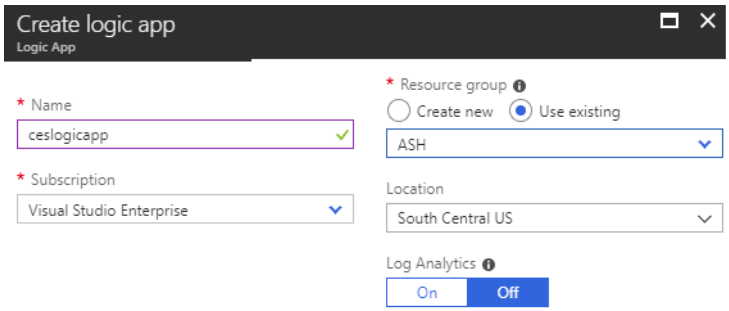
For more information , you can see:

Logic Apps - **https://azure.microsoft.com/en-us/services/logic-apps/**

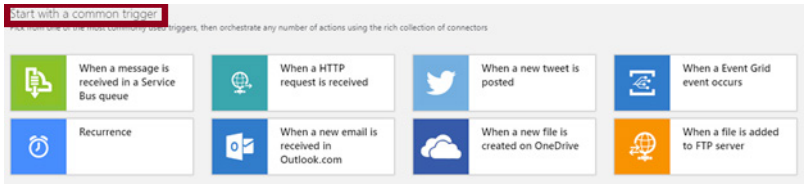
Logic App pricing - **https://azure.microsoft.com/en-us/pricing/details/logic-apps/**

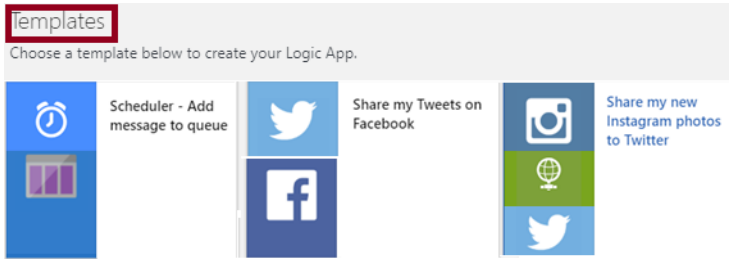
Implementing Logic Apps

To begin using Logic Apps simply create a logic app with Name, Subscription, Resource Group, Location, and Log Analytics (optional).



After Azure deploys your app, the Logic Apps Designer opens and shows a page with commonly used triggers and templates. Logic Apps can be designed end-to-end in the browser. Start with a trigger, including things like a simple schedule, or whenever a tweet appears about your company. Then orchestrate any number of actions using the rich gallery of connectors.





✓ Take a minute to create a Logic App in the portal and browse the triggers and templates that are available. Does anything seem of interest to you?

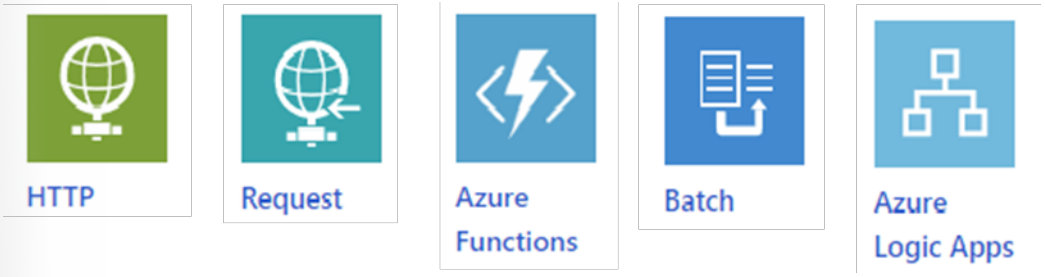
For more information, you can see:

Logic Apps documentation - **https://docs.microsoft.com/en-us/azure/logic-apps/**

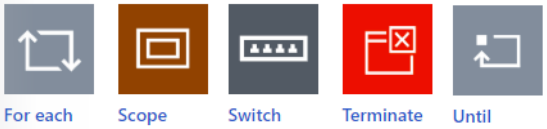
Built-In Triggers and Actions

Logic Apps provides built-in triggers and actions, so you can create schedule-based workflows, help your logic apps communicate with other apps and services, control the workflow through your logic apps, and manage or manipulate data.

In the previous example we used the Schedule built-in. With the Recurrence trigger, you can set a date and time for starting the recurrence and a recurrence schedule for performing tasks. Other built-in triggers include: HTTP, Request, Azure Functions, Batch, and other Azure Logic apps.



There are also built-in actions for structuring and controlling the actions in your logic app's workflow. For example, you could insert a Condition to evaluate a condition and run different actions based on whether the condition is true or false. Other built-in actions are: For each, Scope, Switch, Terminate, and Until.



The Logic App designer will automatically apply the built-ins when creating your workflow, but you can customize the workflow at any time.

For more information, you can see:

Built-ins - **https://docs.microsoft.com/en-us/azure/connectors/apis-list#built-ins**

Managed Connectors

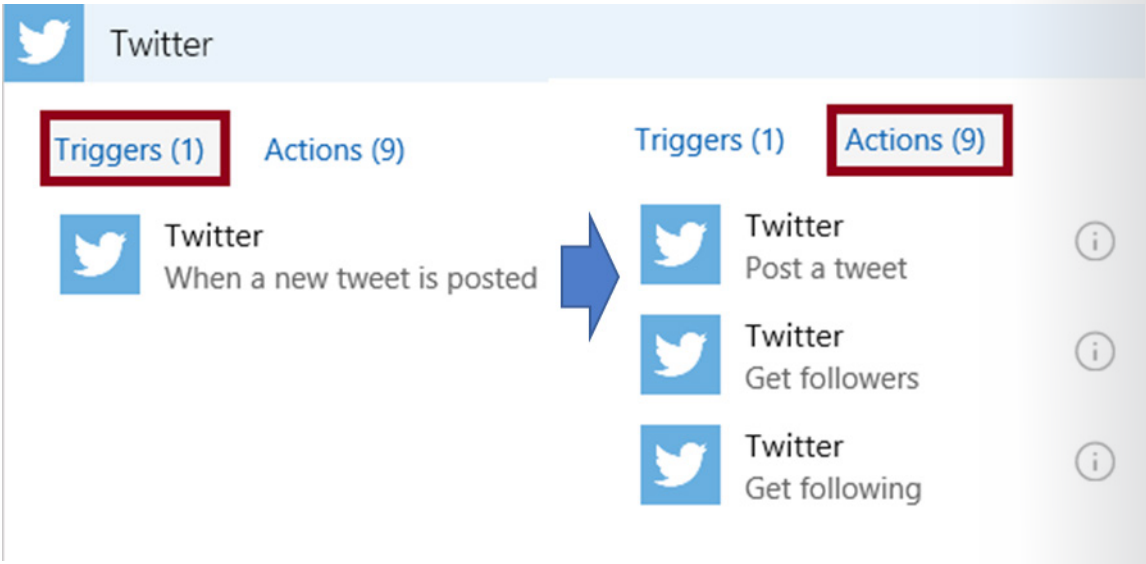
Managed connectors play an integral part when you create automated workflows with Azure Logic Apps. By using connectors in your logic apps, you expand the capabilities for your on-premises and cloud apps to perform tasks with the data that you create and already have.

Logic Apps offers ~200+ connectors, including:

- **Managed API connectors**[49]. This includes Azure Blob Storage, Office 365, Dynamics, Power BI, OneDrive, Salesforce, and SharePoint Online.
- **On-premises connectors**[50]. This includes SQL Server, SharePoint Server, Oracle DB, Twitter, Salesforce, Facebook, and file shares.
- **Integration account connectors**[51]. Available when you create and pay for an integration account, these connectors transform and validate XML, encode, and decode flat files, and process business-to-business (B2B) messages with AS2, EDIFACT, and X12 protocols.
- **Enterprise connectors**[52]. Provide access to enterprise systems such as SAP and IBM MQ for an additional cost.

Whenever you select to include a managed connector, Logic Apps has already configured triggers and actions for that product. For example, with Twitter whenever a new tweet is posted you can get those who are following the tweet.

49 https://docs.microsoft.com/en-us/azure/connectors/apis-list
50 https://docs.microsoft.com/en-us/azure/connectors/apis-list
51 https://docs.microsoft.com/en-us/azure/connectors/apis-list
52 https://docs.microsoft.com/en-us/azure/connectors/apis-list



✓ Are you planning to use any of these connectors?

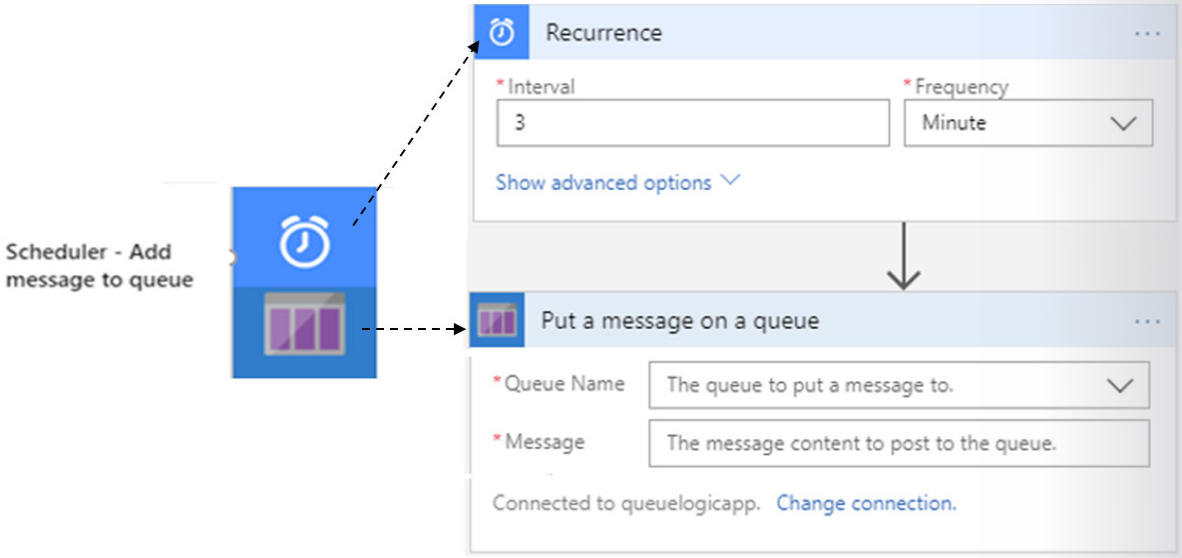
For more information, you can see:

Connectors for Azure Logic Apps - **https://docs.microsoft.com/en-us/azure/connectors/apis-list**

Create a streaming customer insights dashboard with Azure Logic Apps and Azure Functions - **https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-scenario-social-serverless**

Logic App Example

One of the easiest Logic App templates to understand is the Scheduler – Add message to queue. Let's take a closer look at this template. The template has two parts: Recurrence and Put a message on a queue. Recurrence is where you configure the schedule for putting messages on the queue. Putting a message on the queue is where you select the Azure queue you want to use. This template also has error handling (not shown) where you can configure a second queue for messages related to that.



The Logic App designer provides these steps in an easy to use format. The designer steps you through the process of building a workflow. No coding is required. When everything has been validated you can save and run the workflow.

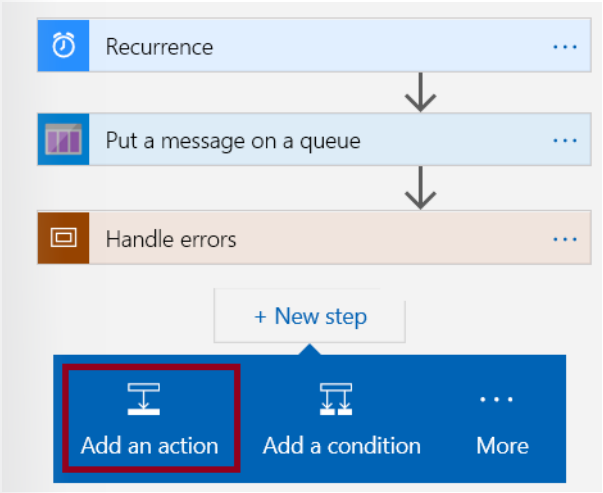
✓ Take a few minutes to create an Azure queue and then use this template to populate the queue. Check to ensure queue messages are arriving on the schedule you configure.

For more information, you can see:

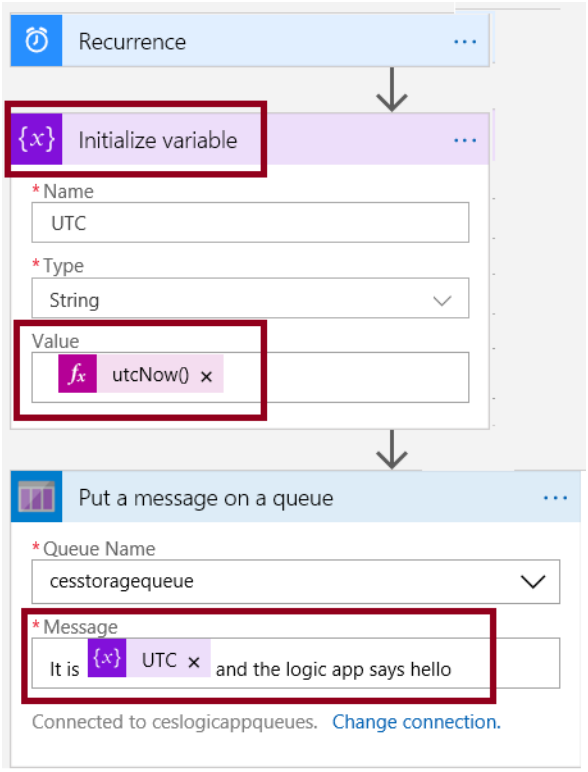
Create and schedule regularly running tasks with Azure Logic Apps - **https://docs.microsoft.com/en-us/azure/connectors/connectors-native-recurrence**

Logic App Example (cont.)

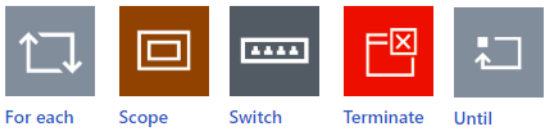
Let's continue with our previous example and customize the workflow. Here is what you should have so far. Click New Step and then Add an action. Scroll through all the different connectors and actions that are available.



In the Actions area select Variable (Initialize Variable). Give your variable a name, String type, and then search the Expressions for UtcNow( ). That will set the value of the variable to the current timestamp. After creating Initialize Variable move it up in the workflow, between Recurrence and Put message on queue. Lastly, add the variable to your message.



✓ Take a few minutes to try this and experiment with actions and using the designer. Be sure to check the Azure queue and make sure the message is working.



Practice: Logic App Workflow (Advanced)

If you are up for a challenge try the **Check traffic with a scheduler-based logic app**[53]. In this practice you will learn how to:

- Create a logic app.
- Add a scheduler trigger.
- Get the travel time for a route.
- Create a variable to store the travel time.
- Add a condition to compare the travel time with the limit.

53 https://docs.microsoft.com/en-us/azure/logic-apps/tutorial-build-schedule-recurring-logic-app-workflow

- Send email when the travel time is exceeded.
- Run and monitor your logic app.

✓ If this practice does not appeal to you, there are other choices in the reference links.

For more information, you can see:

Manage mailing list requests with a logic app - **https://docs.microsoft.com/en-us/azure/logic-apps/tutorial-process-mailing-list-subscriptions-workflow**

Process emails and attachments with a logic app - **https://docs.microsoft.com/en-us/azure/logic-apps/tutorial-process-email-attachments-workflow**

Review Questions

Module 2 Review Questions

Web Apps Features

Your organization plans to host a corporate website in Azure. Traffic to the website is expected to vary at different times of the year.

The website must scale based on demand.

What benefits can you realize by hosting the web as a Web App in Azure?

Suggested Answer ↓

A web app is the compute resources that Azure provides for hosting a website or web application.

The compute resources may be on shared or dedicated virtual machines (VMs), depending on the pricing tier that you choose. Your application code runs in a managed VM that is isolated from other customers.

Web Apps Features (Deployment)

Your organization plans to host a corporate website in Azure. The website includes an e-commerce solution which must be available to customers at all time.

Developers continuously create new solutions to improve customer experiences.

You need to ensure that you can deploy code without affecting user access or any orders in progress.

What should you use? What benefits you will realize?

Suggested Answer ↓

When you deploy your web app, mobile back end, and API app to App Service, you can deploy to a separate deployment slot instead of the default production slot when running in the Standard or Premium App Service plan mode.

Deployment slots are live apps with their own hostnames. App content and configuration elements can be swapped between two deployment slots.

Event Grid

Your organization develops an Azure-based photo management and editing app. You are designing a solution that implements Event Grid to run image analysis code each time a new photo is uploaded.

What are the benefits and limitations of Event Grid?

Suggested Answer ↓

Serverless computing is driven by the reaction to events and triggers happening in near-real-time—in the cloud. As a fully managed service, server management and capacity planning are invisible to the developer and billing is based just on resources consumed or the actual time your code is running.

Serverless computing has many advantages. Here are a few:

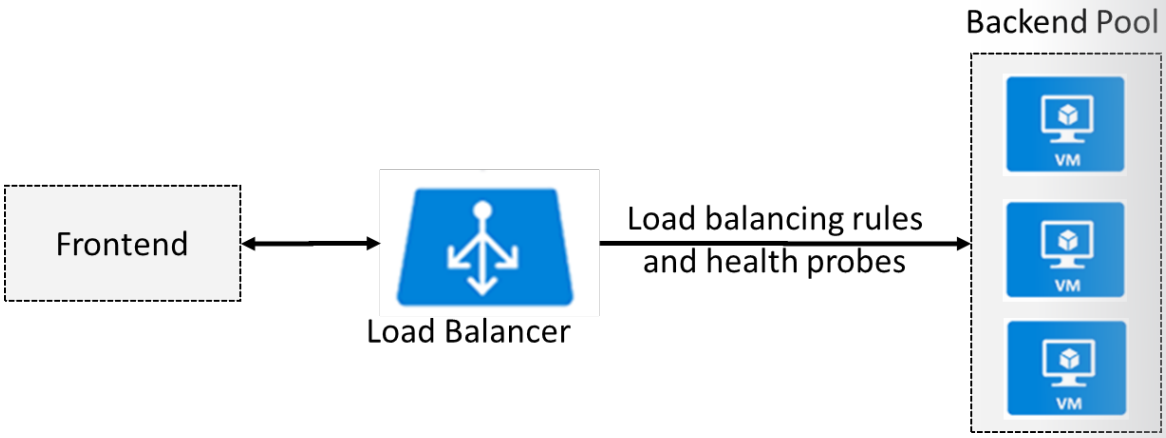
• Benefit from a fully managed service. Organizations can relieve their teams from the burden of managing servers. By using fully managed services, developers focus on application business logic and avoid administrative tasks. With serverless architecture developers simply deploy their code, and it runs with high availability.
• Scale flexibly. Serverless compute scales from nothing to handling tens of thousands of concurrent functions almost instantly (within seconds), to match any workload, and without requiring scale configuration.
• Only pay for the resources used. With serverless architecture, your organization only pays for the time the application code is running. Serverless computing is event-driven, and resources are allocated as soon as they are triggered by an event. You are only charged for the time and resources it takes to execute the application code—through sub-second billing.

Module 3 Module Implementing Advanced Virtual Networking

Azure Load Balancer

Load Balancer

The Azure Load Balancer delivers high availability and network performance to your applications. It is an OSI Layer 4 (TCP and UDP) load balancer that distributes inbound traffic to backend resources using load balancing rules and health probes. Load balancing rules determine how traffic is distributed to the backend. Health probes ensure the resources in the backend are healthy.



The Load Balancer can be used for inbound as well as outbound scenarios and scales up to millions of flows for all TCP and UDP applications.

✓ Keep this diagram in mind since it covers the four components that must be configured for your load balancer: Frontend IP configuration, Backend pools, Health probes, and Load balancing rules.

For more information, you can see:

Load Balancer - **https://azure.microsoft.com/en-us/services/load-balancer/**

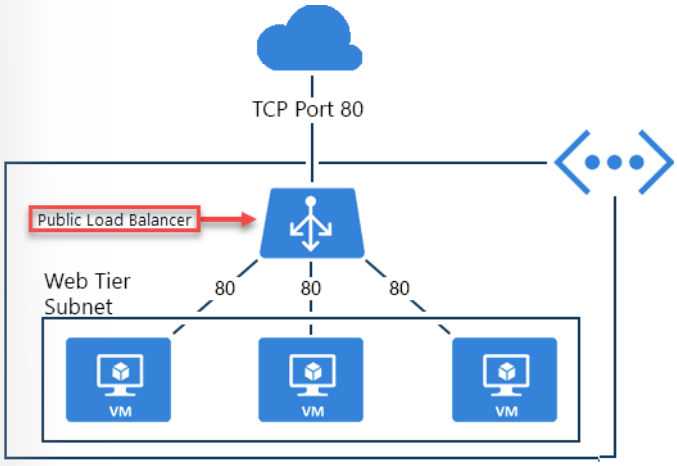
Load Balancer documentation - **https://docs.microsoft.com/en-us/azure/load-balancer/**

Public load balancer

There are two types of load balancers: **public** and **internal**.

A public load balancer maps the public IP address and port number of incoming traffic to the private IP address and port number of the VM, and vice versa for the response traffic from the VM. By applying load-balancing rules, you can distribute specific types of traffic across multiple VMs or services. For example, you can spread the load of incoming web request traffic across multiple web servers.

The following figure shows internet clients sending webpage requests to the public IP address of a web app on TCP port 80. Azure Load Balancer distributes the requests across the three VMs in the load-balanced set.



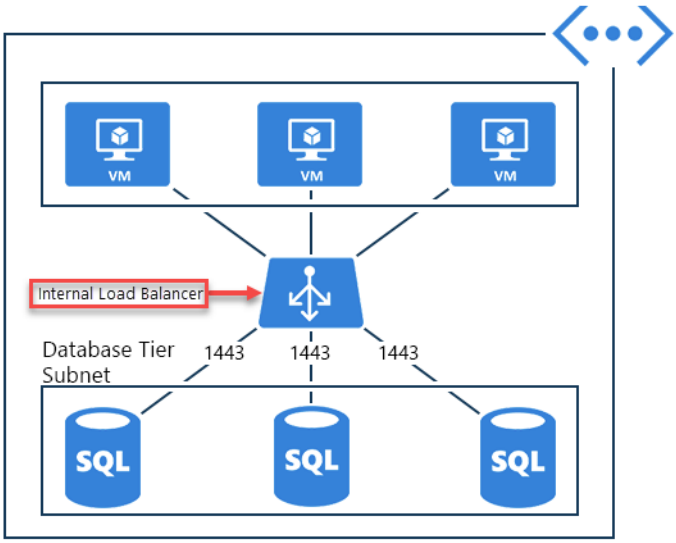
For more information, you can see:

Public load balancer - **https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview#a-name–publicloadbalancerapublic-load-balancer**[1]

Internal load balancer

An internal load balancer directs traffic only to resources that are inside a virtual network or that use a VPN to access Azure infrastructure. Frontend IP addresses and virtual networks are never directly exposed to an internet endpoint. Internal line-of-business applications run in Azure and are accessed from within Azure or from on-premises resources. For example, an internal load balancer could receive database requests that need to be distributed to backend SQL servers.

1 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview



An internal load balancer enables the following types of load balancing:

- **Within a virtual network**. Load balancing from VMs in the virtual network to a set of VMs that reside within the same virtual network.
- **For a cross-premises virtual network**. Load balancing from on-premises computers to a set of VMs that reside within the same virtual network.
- **For multi-tier applications**. Load balancing for internet-facing multi-tier applications where the backend tiers are not internet-facing. The backend tiers require traffic load-balancing from the internet-facing tier.
- **For line-of-business applications**. Load balancing for line-of-business applications that are hosted in Azure without additional load balancer hardware or software. This scenario includes on-premises servers that are in the set of computers whose traffic is load-balanced.

✓ Can you see how a public load balancer could be placed in front of the internal load balancer to create a multi-tier application.

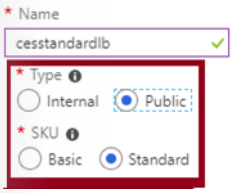
For more information, you can see:

Internal load balancer - **https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview#a-name–internalloadbalancera-internal-load-balancer**[2]

Load Balancer SKUs

When you create an Azure Load Balancer you will select for the type (Internal or Public) of load balancer. You will also select the SKU. The load balancer supports both Basic and Standard SKUs, each differing in scenario scale, features, and pricing. The Standard Load Balancer is the newer Load Balancer product with an expanded and more granular feature set over Basic Load Balancer. It is a superset of Basic Load Balancer.

2 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview



Here is some general information about the SKUs.

- SKUs are not mutable. You may not change the SKU of an existing resource.
- A standalone virtual machine resource, availability set resource, or virtual machine scale set resource can reference one SKU, never both.
- A Load Balancer rule cannot span two virtual networks. Frontends and their related backend instances must be in the same virtual network.
- There is no charge for the Basic load balancer. The Standard load balancer is charged based on number of rules and data processed. Read more at the reference link.
- Load Balancer frontends are not accessible across global virtual network peering.
- ✓ New designs and architectures should consider using Standard Load Balancer.

For more information, you can see:

Why use Standard Load Balancer - **https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-standard-overview#why-use-standard-load-balancer**[3]

Load Balancer limits - **https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits#load-balancer**[4]

Load Balancer SKU comparison - **https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview#skus**[5]

Load Balancer pricing - **https://azure.microsoft.com/en-us/pricing/details/load-balancer/**

Backend Pool

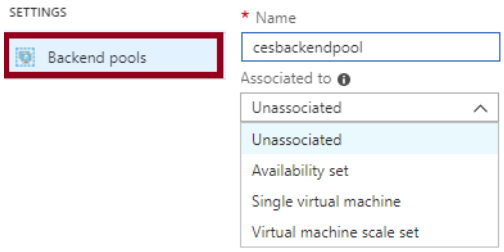
To distribute traffic, a back-end address pool contains the IP addresses of the virtual NICs that are connected to the load balancer.

How you configure the backend pool depends on whether you are using the Standard or Basic SKU.

3 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-standard-overview
4 https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits
5 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview

-	Standard SKU	Basic SKU
Backend pool endpoints	Any VM in a single virtual network, including a blend of VMs, availability sets, and VM scale sets.	VMs in a single availability set or VM scale set.

Backend pools are configured from the Backend Pool blade. For the Standard SKU you can connect to an Availability set, single virtual machine, or a virtual machine scale set.



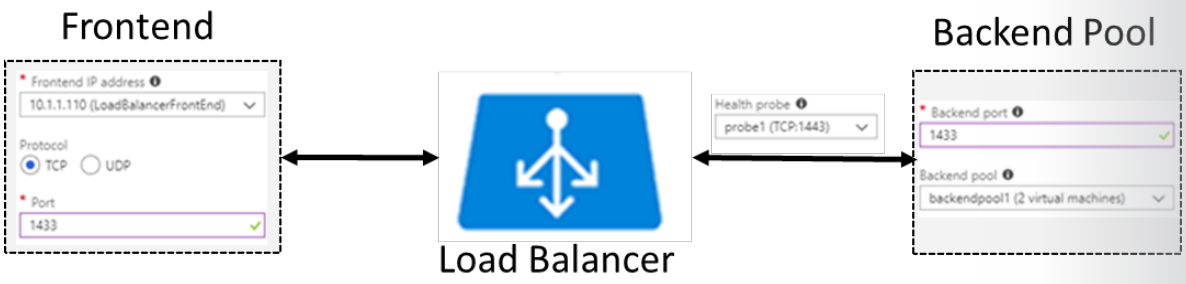
✓ In the Standard SKU you can have up to 1000 instances in the backend pool. In the Basic SKU you can have up to 100 instances.

For more information, you can see:

Create a backend address pool - **https://docs.microsoft.com/en-us/azure/load-balancer/quick-start-load-balancer-standard-public-portal#create-a-backend-address-pool**[6]

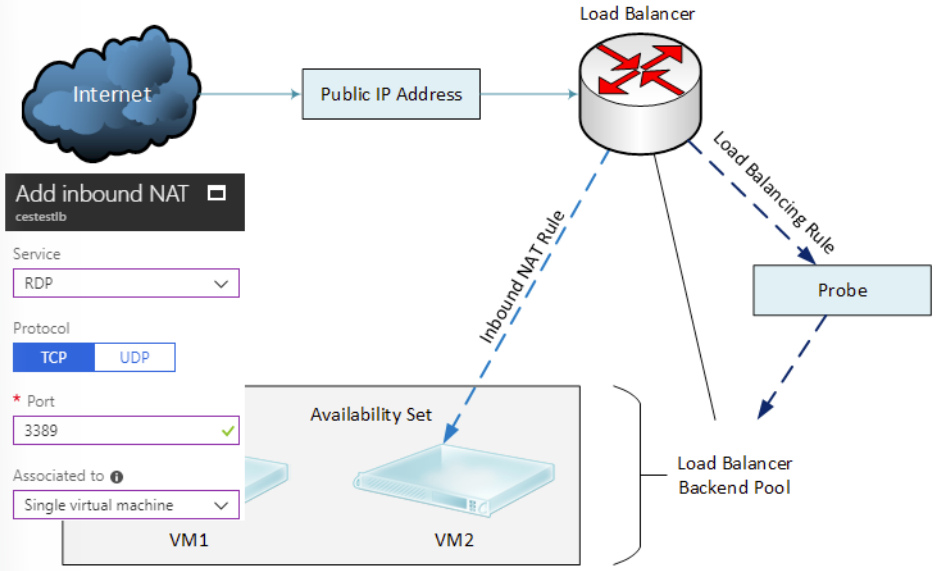
Load Balancer Rules

A load balancer rule is used to define how traffic is distributed to the backend pool. The rule maps a given frontend IP and port combination to a set of backend IP addresses and port combination. To create the rule the frontend, backend, and health probe information should already be configured. Here is a rule that passes frontend TCP connections to a set of backend SQL (port 1433) servers. The rule uses a health probe that checks on TCP 1443.



Load balancing rules can be used in combination with NAT rules. For example, you could NAT TCP from the load balancer's public address to TCP 3389 on a specific virtual machine. This allows remote desktop access from outside of Azure. Notice in this case, the NAT rule is explicitly attached to a VM (or network interface) to complete the path to the target; whereas a Load Balancing rule need not be.

6 https://docs.microsoft.com/en-us/azure/load-balancer/quickstart-load-balancer-standard-public-portal



✓ Can you see the difference between load balancing rules and NAT rules? Remember, this approach should only be used when you need connectivity from the Internet. Most normal communications would occur from on-premises to Azure connections such as site-to-site VPN and ExpressRoute.

For more information, you can see:

Create a load balancer rule - **https://docs.microsoft.com/en-us/azure/load-balancer/quick-start-load-balancer-standard-public-portal#create-a-load-balancer-rule**[7]

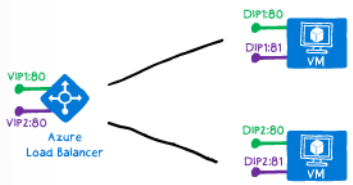
Multiple Frontends

Azure Load Balancer allows you to load balance services on multiple ports, multiple IP addresses, or both. You can use public and internal load balancer definitions to load balance flows across a set of VMs. Adding multiple frontends is incremental to a single frontend configuration.

When you define an Azure Load Balancer, frontend and backend pool configurations are connected with rules. There are two types of rules:

1. The default rule with no backend port reuse
2. The Floating IP rule where backend ports are reused

Rule type 1: No backend port reuse



7 https://docs.microsoft.com/en-us/azure/load-balancer/quickstart-load-balancer-standard-public-portal

In this scenario, the default rule, the frontends are configured with values for IP address, protocol, and port. The DIP is the destination of the inbound flow. In the backend pool, each VM exposes the desired service on a unique port on a DIP. This service is associated with the frontend through a rule definition.

Each rule must produce a flow with a unique combination of destination IP address and destination port. By changing the destination port of the flow, multiple rules can distribute flows to the same DIP on different ports.

✓ This topic continues on the next page.

Multiple Frontends (Rule 2)

Multiple Frontends

Rule type #2: backend port reuse by using Floating IP

If you want to reuse the backend port across multiple rules, you must enable Floating IP in the rule definition. Floating IP refers to the portion of what is known as Direct Server Return (DSR). DSR consists of two parts:

- A flow topology
- IP address mapping scheme.

At a platform level, Azure Load Balancer always operates in a DSR flow topology regardless of whether Floating IP is enabled or not. This means that the outbound part of a flow is always correctly rewritten to flow directly back to the origin.

As opposed to the traditional load balancing mapping scheme used by the default rule, enabling Floating IP changes the IP address mapping scheme to allow for additional flexibility as explained below.

The following diagram illustrates this configuration:

<img src="../../Linked_Image_Files/AZ-101.3_Implementing_Advanced_Virtual_Networking_image10.png" alt="Diagram showing the rule type where a backend port is reused across multiple rules. This shows how a Virtual IP is used to “float” between the frontend and backends." title="">

For this scenario, every VM in the backend pool has three network interfaces:

- **DIP**. A Virtual NIC associated with the VM (IP configuration of Azure's NIC resource)
- **Frontend 1**. A loopback interface within guest OS that is configured with IP address of Frontend 1
- **Frontend 2**. A loopback interface within guest OS that is configured with IP address of Frontend 2

If we define rules mapping the frontend to the backend pool, the mapping in the load balancer would include frontend IP address, protocol, destination and ports. The destination of the inbound flow is the frontend IP address on the loopback interface in the VM. By changing the destination IP address, you can enable port reuse on the same VM.

✓ Make sure to take into account **limitations to using load balancers with multiple frontends**[8].

✓ Can you think of ways in which extending your load balancer to multiple ports and IP addresses would benefit your network configuration?

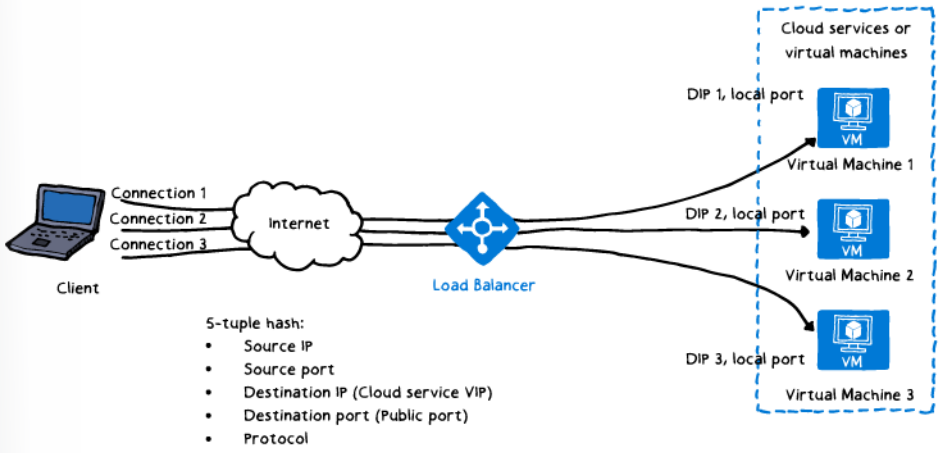
For more information, see:

Multiple Frontends for Azure Load Balancer - **https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-multivip-overview**

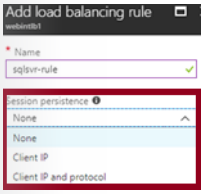
8 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-multivip-overview

Session Persistence

By default, Azure Load Balancer distributes network traffic equally among multiple VM instances. The load balancer uses a 5-tuple (source IP, source port, destination IP, destination port, and protocol type) hash to map traffic to available servers. It provides stickiness only within a transport session.



Session persistence specifies how traffic from a client should be handled. The default behavior (None) is that successive requests from a client may be handled by any virtual machine. You can change this behavior.



- **Client IP** specifies that successive requests from the same client IP address will be handled by the same virtual machine.
- **Client IP and protocol** specifies that successive requests from the same client IP address and protocol combination will be handled by the same virtual machine.

✓ Keeping session persistence information is very important in applications that use a shopping cart. Can you think of any other applications?

For more information, you can see:

Configure the distribution mode for Azure Load Balancer - **https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-distribution-mode**

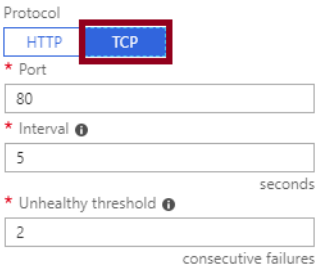
Health Probes

A health probe allows the load balancer to monitor the status of your app. The health probe dynamically adds or removes VMs from the load balancer rotation based on their response to health checks. When a probe fails to respond, the load balancer stops sending new connections to the unhealthy instances.

There are two main ways to configure health probes: **HTTP** and **TCP**.

HTTP custom probe****. The load balancer regularly probes your endpoint (every 15 seconds, by default). The instance is healthy if it responds with an HTTP 200 within the timeout period (default of 31 seconds). Any status other than HTTP 200 causes this probe to fail. You can specify the port (Port), the URI for requesting the health status from the backend (URI), amount of time between probe attempts (Interval), and the number of failures that must occur for the instance to be considered unhealthy (Unhealthy threshold).

TCP custom probe. This probe relies on establishing a successful TCP session to a defined probe port. If the specified listener on the VM exists, the probe succeeds. If the connection is refused, the probe fails. You can specify the Port, Interval, and Unhealthy threshold.



✓ There is also a guest agent probe. This probe uses the guest agent inside the VM. It is not recommended when HTTP or TCP custom probe configurations are possible.

For more information, you can see:

HTTP custom probe - **https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-custom-probe-overview#http-custom-probe**[9]

TCP custom probe - **https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-custom-probe-overview#tcp-custom-probe**[10]

Demonstration: Network Load Balancer

9 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-custom-probe-overview
10 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-custom-probe-overview

Practice: Standard Load Balancer

Take a few minutes to try the **QuickStart: Create a Standard Load Balancer to load balance VMs using the Azure portal**[11]. This QuickStart shows you how to load balance VMs using a Standard Load Balancer. Specifically, you will learn how to:

- Create a public load balancer.
- Create backend servers (virtual network, virtual machines, NSG rules, .
- Create load balancer resources (backend address pool, health probe, load balancer rules).
- Test the load balancer.

Also, be sure to try the **QuickStart: Create a Standard Load Balancer using Azure PowerShell**[12]. This QuickStart shows you how to configure the load balancer with PowerShell.

✓ If you prefer, use the reference link to try the CLI version of this QuickStart.

For more information, you can see:

QuickStart: Create a Standard Load Balancer to load balance VMS using Azure CLI 2.0 - **https://docs.microsoft.com/en-us/azure/load-balancer/quickstart-load-balancer-standard-public-cli**

Practice: Load Balancer ARM Deployments

Take a few minutes to try at least one of the QuickStart templates to deploy a load balancer. For example, **2 VMs in a Load Balancer and load balancing rules**[13]. This template allows you to create 2 Virtual Machines under a Load balancer and configure a load balancing rule on Port 80. This template also deploys a Storage Account, Virtual Network, Public IP address, Availability Set and Network Interfaces. Another example is **2 VMs in VNET - Internal Load Balancer and LB rules**[14].

For more information, you can see:

Azure QuickStart Templates - **https://azure.microsoft.com/en-us/resources/templates/**

11 https://docs.microsoft.com/en-us/azure/load-balancer/quickstart-load-balancer-standard-public-portal
12 https://docs.microsoft.com/en-us/azure/load-balancer/quickstart-create-standard-load-balancer-powershell
13 https://azure.microsoft.com/en-us/resources/templates/201-2-vms-loadbalancer-lbrules/
14 https://azure.microsoft.com/en-us/resources/templates/201-2-vms-internal-load-balancer/

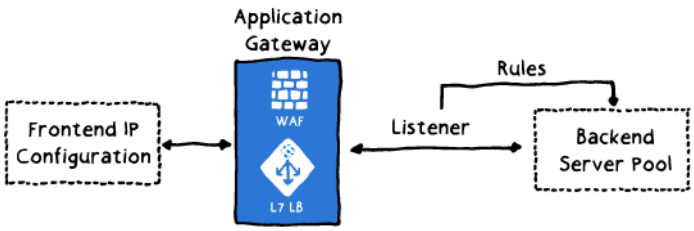
Azure Application Gateway

Application Gateway Components

Azure Application Gateway is a web traffic load balancer that enables you to manage traffic to your web applications.

Traditional load balancers operate at the transport layer (OSI layer 4 - TCP and UDP) and route traffic based on source IP address and port, to a destination IP address and port. An example is the Azure Load Balancer. But the Application Gateway is an OSI layer 7 load balancer, so you can be even more specific in routing your network traffic.

The Application Gateway has many of the same components and configuration tasks as the Load Balancer. But, the terminology is a little different, so let's review.



Frontend IP configuration. This configured port is a public port that is opened on the Application Gateway. Traffic hits this port, and then gets redirected to one of the backend servers.

Backend server pool. The list of IP addresses of the backend servers. The IP addresses listed should either belong to the virtual network subnet or should be a public IP/VIP. Every pool has settings like port, protocol, and cookie-based affinity. These settings are tied to the pool and are applied to all servers within the pool.

Listener. The listener has a front-end port, a protocol (HTTP or HTTPS), , and the SSL certificate name (optional).

Rule. The rule binds the listener and the backend server pool and defines which backend server pool the traffic should be directed to when it hits a listener.

Web application firewall (WAF). A WAF is provided as part of the Application gateway. WAF integration makes security management much simpler and can react faster to a security threat by patching a known vulnerability at a central location versus securing each of the individual web applications.

✓ Application gateway can be configured as an Internet-facing gateway, internal-only gateway, or a combination of both.

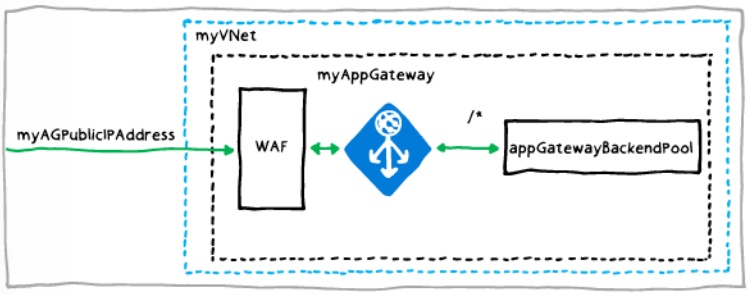
For more information, you can see:

Application Gateway - **https://azure.microsoft.com/en-us/services/application-gateway/**

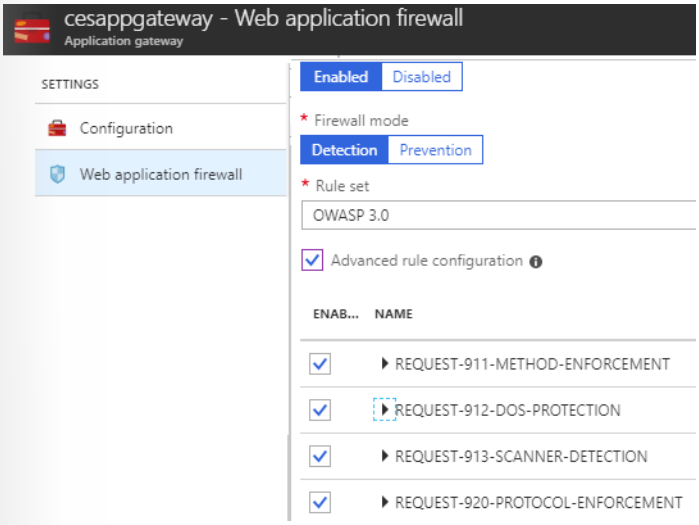
Web Application Firewall

The Azure Application Gateway WAF provides protection for web applications. The WAF uses **OWASP**[15] rules to protect your application. These rules include protection against attacks such as SQL injection, cross-site scripting attacks, and session hijacks.

15 https://www.owasp.org/index.php/Category:OWASP_ModSecurity_Core_Rule_Set_Project



By default, all OWASP rules are applied to your Application gateway. Some rules can cause false positives and block real traffic. So, you can edit the rule set and disable any rules that are not appropriate. You can disable an entire rule group or specific rules under one or more rule groups.



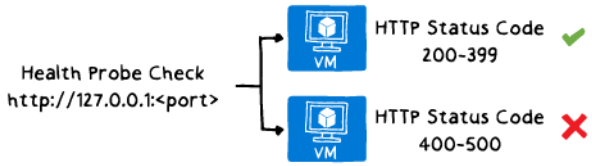
For more information, you can see:

Customize web application firewall rules through the Azure portal - **https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-customize-waf-rules-portal**

Health Probes

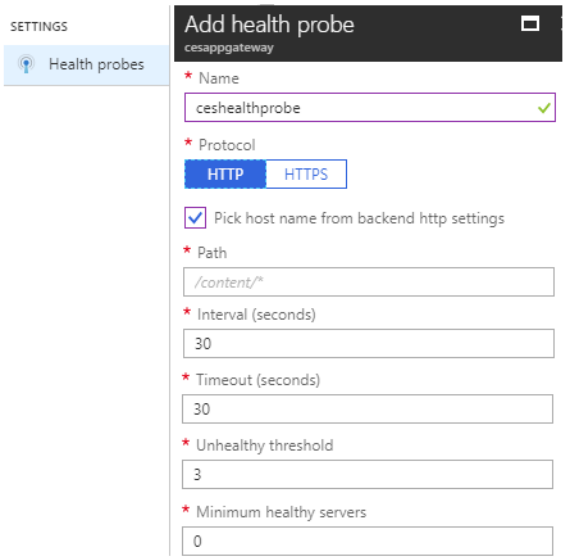
By default, Azure Application Gateway monitors the health of all resources in its backend pool and automatically removes any resource considered unhealthy from the pool. Application Gateway continues to monitor the unhealthy instances and adds them back to the healthy backend pool once they become available and respond to health probes.

For example, if you have two backend servers receiving HTTP network traffic on port 80, the default health probe checks every 30 seconds for a healthy HTTP response code between 200-399.



Implementation

If you need to configure the health probe to go to a custom URL or modify any other settings, you must configure a custom health probe. Custom probes are useful for applications that have a specific health check page or for applications that do not provide a successful response on the default web application. Custom probes allow you to have more control over health monitoring. When using custom probes, you can configure the probe interval, the URL and path to test, and how many failed responses to accept before marking the back-end pool instance as unhealthy.



✓ Do you think you will need a custom health probe or will the default health probe be acceptable? Explore the reference links for more information.

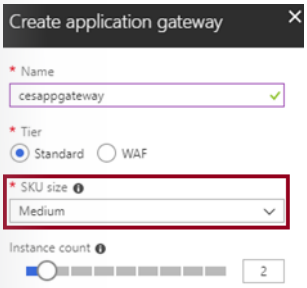
For more information, you can see:

Create a custom probe for Application Gateway by using the portal - **https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-create-probe-portal**

Create a custom probe for Azure Application Gateway by using PowerShell for Azure Resource Manager - **https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-create-probe-ps**

Application Gateway Sizing

When you create an Application Gateway there are three SKU sizes: Small, Medium, and Large. Small instance sizes are only intended for development and testing scenarios.



Each SKU provides different performance benefits. The following table shows the average performance throughput based on the backend page size. These are estimates, with SSL offload enabled.

Average back-end page response size	Small	Medium	Large
6KB	7.5 Mbps	13 Mbps	50 Mbps
100KB	35 Mbps	100 Mbps	200 Mbps

There are also **Application Gateway service limits**[16]. For example, you can currently create up to 50 application gateways per subscription, 20 frontend ports, and 20 backend pools with 100 backend servers per pool. The caveat is that all such metrics can change over time, so be sure to consult the documentation.

✓ We recommend a minimum instance count of two for production workloads. Azure distributes instances across update and fault domains to ensure that all instances do not fail at the same time. Your virtual network and public IP address must be in the same location as your gateway.

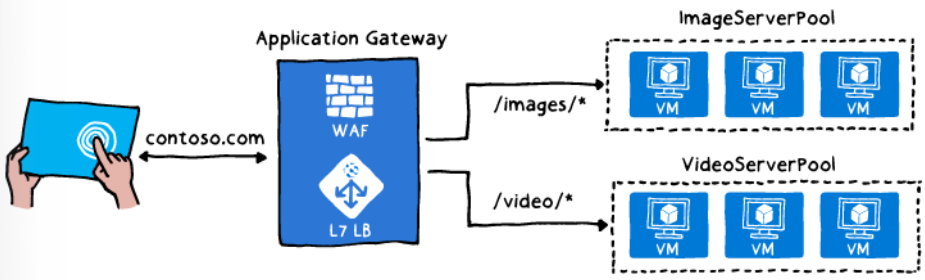
For more information, you can see:

Application Gateway pricing - **https://azure.microsoft.com/en-us/pricing/details/application-gateway/**

Application Gateway FAQ - **https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-faq**

Path-Based Routing

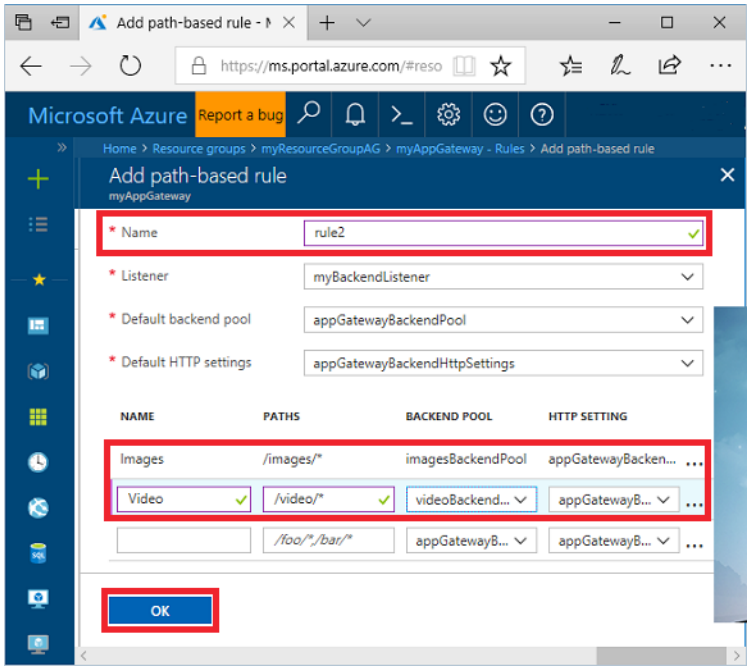
You can route traffic based on the incoming URL. So, if /images is in the incoming URL, you can route traffic to a backend pool of servers configured for images. If /video is in the URL, that traffic is routed to another backend pool of servers optimized for video streaming.



16 https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits?toc=%2fazure%2fapplication-gateway%2ftoc.json

Implementation

To configure a path-based rule you must define the path pattern to match. For example, /images/* or /video/*.



✓ Do you have a need for path-based rules? Be sure to try this for yourself using the reference links below.

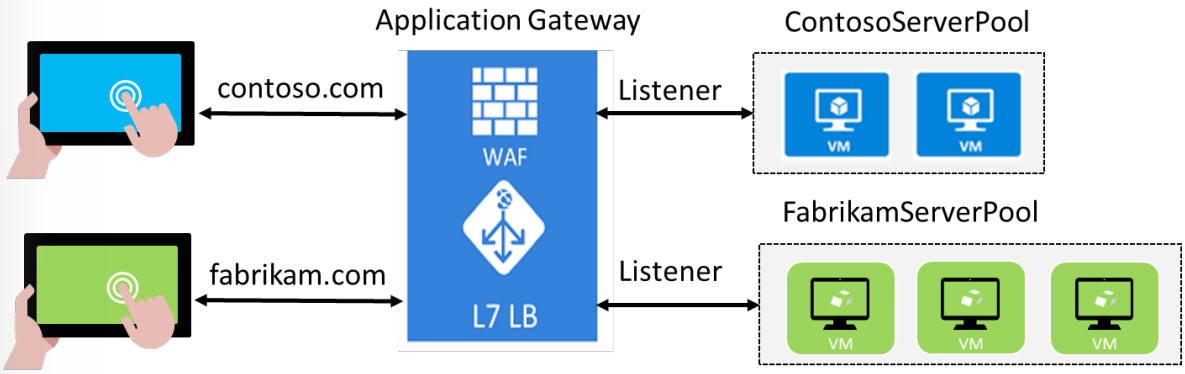
For more information, you can see:

Create an application gateway with path-based routing rules using the Azure portal - **https://docs.microsoft.com/en-us/azure/application-gateway/create-url-route-portal**

Route web traffic based on the URL using Azure PowerShell - **https://docs.microsoft.com/en-us/azure/application-gateway/tutorial-url-route-powershell**

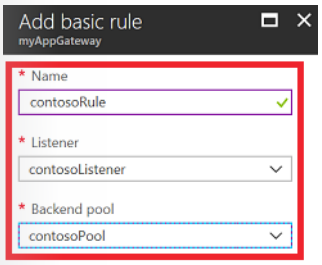
Multiple Site Hosting

Multiple site hosting enables you to configure more than one web site on the same Application Gateway instance. You can add up to 20 web sites to one Application Gateway. Each web site can be directed to its own backend pool of servers. For example, an Application Gateway can serve traffic for contoso.com and fabrikam.com from two server pools called ContosoServerPool and FabrikamServerPool.



Implementation

As the above diagram shows you will need to configure two backend server pools and two listeners. You will also need two routing rules to connect the listeners to the backend pools.



Rules are processed in the order they are listed. For example, if you have a rule using a basic listener and a rule using a multi-site listener both on the same port, the rule with the multi-site listener must be listed before the rule with the basic listener in order for the multi-site rule to function as expected.

✓ After the application gateway is created with its public IP address, you can get the DNS address and use it to create a CNAME record in your domain. Be sure to try the practical exercises in the reference links.

For more information, you can see:

Create and configure an application gateway to host multiple web sites using the Azure portal - **https://docs.microsoft.com/en-us/azure/application-gateway/create-multiple-sites-portal#create-an-application-gateway**[17]

Create an application gateway that hosts multiple web sites using Azure PowerShell - **https://docs.microsoft.com/en-us/azure/application-gateway/tutorial-multiple-sites-powershell**

Secure Sockets Layer Offload

Application Gateway supports SSL termination at the gateway, after which traffic typically flows unencrypted to the backend servers. This feature allows web servers to be unburdened from costly encryption and decryption overhead.

Implementation

17 https://docs.microsoft.com/en-us/azure/application-gateway/create-multiple-sites-portal

You can use **New-SelfSignedCertificate**[18] to create a self-signed certificate that you upload to the Azure portal when you create the listener for the application gateway.

<img src="../../Linked_Image_Files/AZ-101.3_Implementing_Advanced_Virtual_Networking_image41.png" alt="Screenshot of the Create application gateway settings. The protocol is HTTPS. The port is 443. The Upload PFX certificate is “appgwcert.pfx”. A certificate name and password is provided. " title="">

Once you have uploaded the certificate you can return to this page and renew/upload a new certificate.

✓ Do you have a large application that can benefit by offloading SSL certificates? Be sure to try one of the practices in the links below.

For more information, you can see:

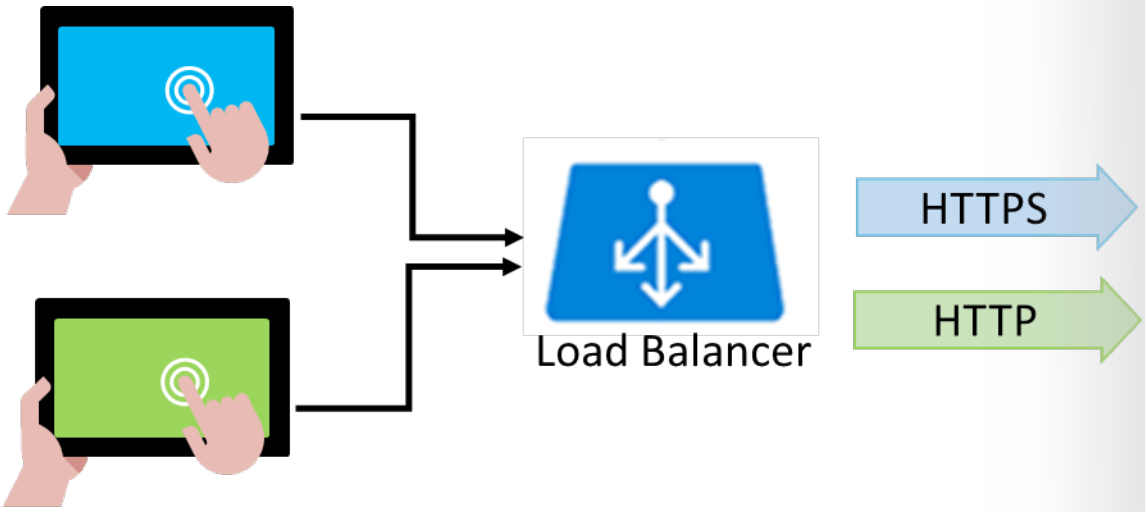
Configure an application gateway with SSL termination using the Azure portal - **https://docs.microsoft.com/en-us/azure/application-gateway/create-ssl-portal**

Create an application gateway with SSL termination using Azure PowerShell - **https://docs.microsoft.com/en-us/azure/application-gateway/tutorial-ssl-powershell**

Redirection and Session Affinity

Application gateway has many useful features. So far, we have addressed the WAF, Path-Based Routing, Multiple Site Hosting, and Secure Socket Layer (SSL) Offload. Here are two more features you might be interested in.

Redirection



Application Gateway supports redirection from one port to another port. This enables HTTP to HTTPS redirection which is a common scenario for all communication between an application and its users occurring over an encrypted path. Application Gateway also supports path-based redirection. This type of redirection enables HTTP to HTTPS redirection only on a specific site area, for example a shopping cart area denoted by /cart/*.

Application Gateway redirection support is not limited to HTTP to HTTPS redirection. This is a generic redirection mechanism, so you can use rules to redirect from and to any port. It also supports redirection to an external site as well.

Session affinity

18 https://docs.microsoft.com/powershell/module/pkiclient/new-selfsignedcertificate

The cookie-based session affinity feature is useful when you want to keep a user session on the same server. By using gateway-managed cookies, the Application Gateway can direct subsequent traffic from a user session to the same server for processing. This is important in cases where session state is saved locally on the server for a user session.

✓ Are any of these features of interest to you?

For more information, you can see:

Application Gateway Features - **https://docs.microsoft.com/en-us/azure/application-gateway/overview#secure-sockets-layer-ssl-termination**[19]

Demonstration: Configuring Application Gateway

Practice: Application Gateway

Take a few minutes to try the **QuickStart: Direct web traffic with Azure Application Gateway - Azure portal**[20]. This QuickStart shows you how to use the Azure portal to quickly create the application gateway with two virtual machines in its backend pool. You then test it to make sure it's working correctly. Specifically, you will learn how to:

- Create an Application Gateway.
- Create backend servers.
- Test the Application Gateway.

✓ If you prefer, use the reference links to try the practice with PowerShell or the CLI.

For more information, you can see:

QuickStart: Direct web traffic with Azure Application Gateway - Azure PowerShell -**https://docs.microsoft.com/en-us/azure/application-gateway/quick-create-powershell**

QuickStart: Direct web traffic with Azure Application Gateway - Azure CLI - **https://docs.microsoft.com/en-us/azure/application-gateway/quick-create-cli**

19 https://docs.microsoft.com/en-us/azure/application-gateway/overview
20 https://docs.microsoft.com/en-us/azure/application-gateway/quick-create-portal

Practice: Web Application Firewall

Take a few minutes to try the **QuickStart: Create an application gateway with a web application firewall using the Azure portal**[21]. This QuickStart shows you how to enable the WAF and configure diagnostics. You then test it to make sure it's working correctly. Specifically, you will learn how to:

- Create an application gateway with WAF enabled
- Create the virtual machines used as backend servers
- Create a storage account and configure diagnostics

✓ If you prefer, use the reference links to try the practice with PowerShell or the CLI.

For more information, you can see:

Enable web application firewall using Azure PowerShell - **https://docs.microsoft.com/en-us/azure/application-gateway/tutorial-restrict-web-traffic-powershell**

Tutorial: Enable web application firewall using the Azure CLI - **https://docs.microsoft.com/en-us/azure/application-gateway/tutorial-restrict-web-traffic-cli**

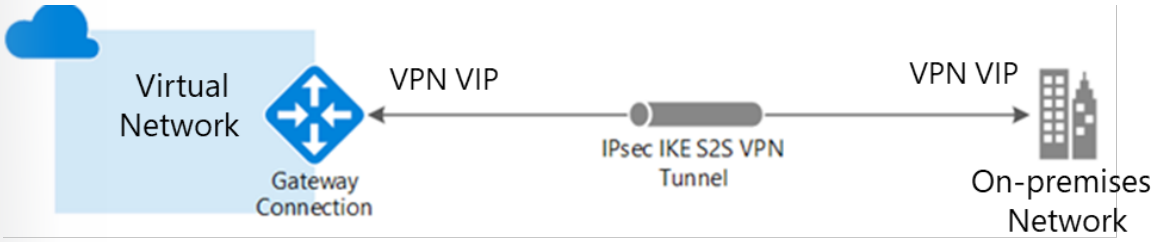
21 https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-web-application-firewall-portal

Site-to-Site VPN Connections

Video: Site-to-Site Connections

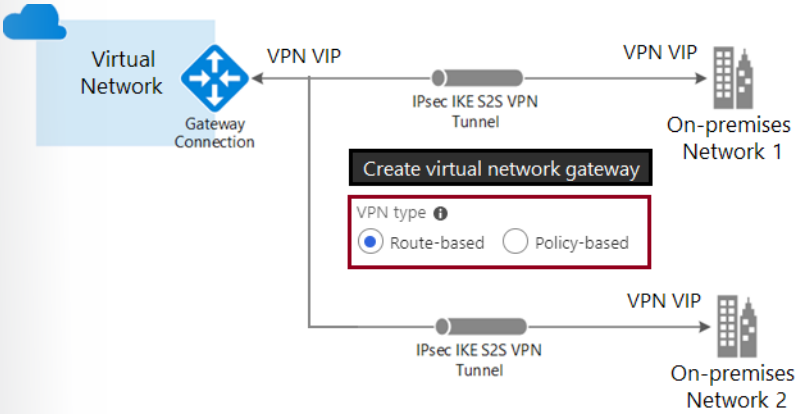
Site-to-Site VPN Connections

A Site-to-Site (S2S) connection is a connection over IPsec/IKE (IKEv1 or IKEv2) VPN tunnel. S2S connections can be used for cross-premises and hybrid configurations. This type of connection requires a VPN device located on-premises that has a public IP address assigned to it.



Multiple Sites

A Multi-site connection is a variation of the Site-to-Site connection. You create more than one VPN connection from your virtual network gateway, typically connecting to multiple on-premises sites. When working with multiple connections, you must use a Route-based VPN. Because each virtual network can only have one VPN gateway, all connections through the gateway share the available bandwidth.



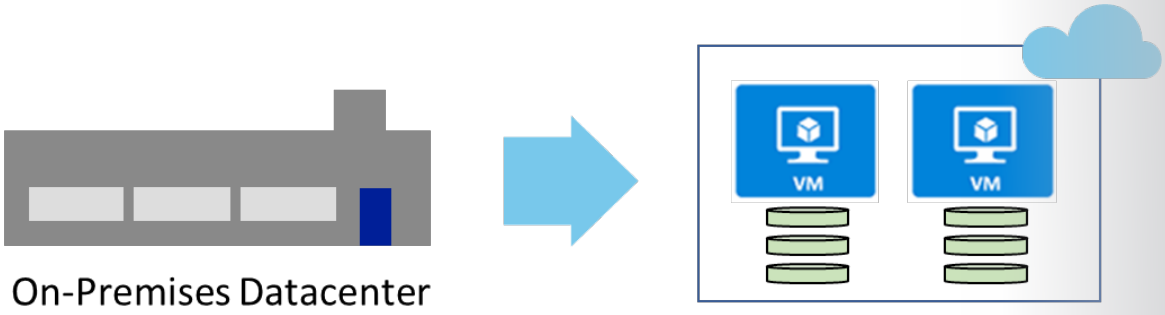
✓ More information on IPsec/IKE cryptographic requirements is available at the reference link.

For more information, you can see:

Site-to-Site and Multi-Site (IPsec/IKE VPN tunnel) - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways#s2smulti**[22]

About cryptographic requirements and Azure VPN gateways - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-compliance-crypto**

Site-to-Site Scenarios



There are many scenarios where Site-to-Site connections can be useful. Here are a few.

Capacity On-Demand

Azure provides capacity on demand. By creating a connection to Azure, more storage or compute resources can easily be brought online. You can extend your on-premises datacenter without purchasing and installing equipment in the datacenter. This scenario includes spawning remote offices.

Strategic Migration

There are many strategic reasons for moving to Azure. Organizations whose core purpose is not related to managing complex datacenter deployments, may want to shed competing interests and focus on improving their core business. They may also want to reduce costs by moving to a pay as you go model. Migrating services is usually faster than responding in-house, especially when you trying to project a global presence.

Disaster Recovery

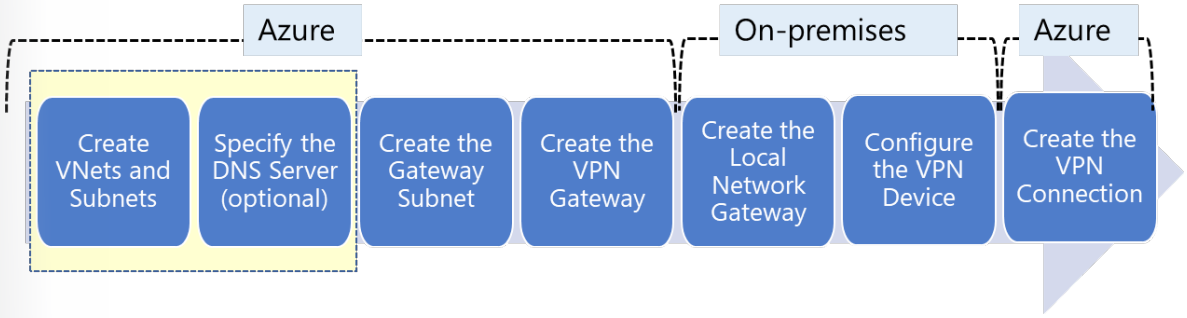
The cloud offers an efficient, cost effective choice for data backup and recovery. Most cloud platforms let you run third-party software for backup and disaster recovery, but with Microsoft these services are fully integrated and easy to turn on, which means you don't have to install and manage a separate product in the cloud.

✓ Do any of these scenarios apply to your organization?

Implementing Site-to-Site VPN

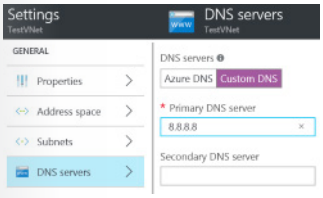
To configure Site-to-Site you must configure both sides of the infrastructure. For example, Azure and on-premises. There are a lot of steps, but we will go through each one. As we do, try to keep the architecture diagrams in mind.

22 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways



Create VNets and subnets. By now you should be familiar with creating virtual networks and subnets. If not, use the reference link. Remember for this VNet to connect to an on-premises location. You need to coordinate with your on-premises network administrator to reserve an IP address range that you can use specifically for this virtual network.

Specify the DNS server (optional). DNS is not required to create a Site-to-Site connection. However, if you want to have name resolution for resources that are deployed to your virtual network, you should specify a DNS server in the virtual network configuration.



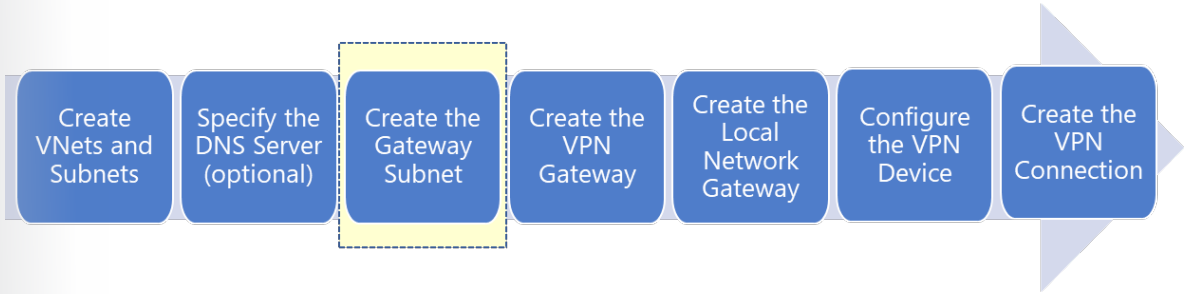
✓ Take time to carefully plan your network configuration. If a duplicate IP address range exists on both sides of the VPN connection, traffic will not route the way you may expect it to. Can you see which steps are completed in Azure and which steps are done on-premises?

For more information, you can see:

Create a virtual network - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-how-to-site-to-site-resource-manager-portal#CreatVNet**[23]

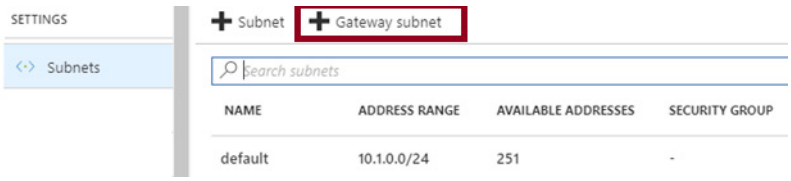
Name resolution for resources in Azure virtual networks - **https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-name-resolution-for-vms-and-role-instances**

Gateway Subnet



23 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal

Before connecting your virtual network to the VPN gateway, you must create a Gateway subnet for the virtual network. When you create a Gateway subnet, VMs are deployed and configured with the required VPN gateway settings. You should not deploy any other resources in this subnet.



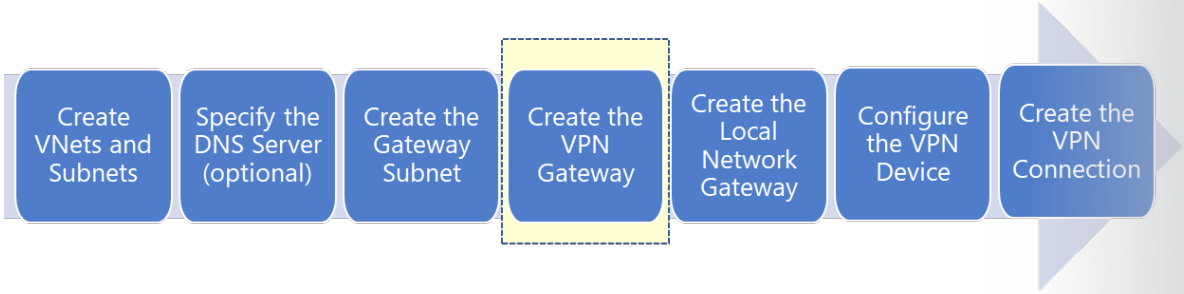
The Gateway subnet prefix for some configurations requires a subnet of /28 or larger to accommodate the number of IP addresses needed in the pool. You may want to create a larger subnet here to accommodate possible future configuration additions. This means the Gateway subnet prefix needs to be /28, /27, /26 etc.

✓ The Gateway subnet must be named GatewaySubnet and you should never associate a Network Security Group with this subnet.

For more information, you can see:

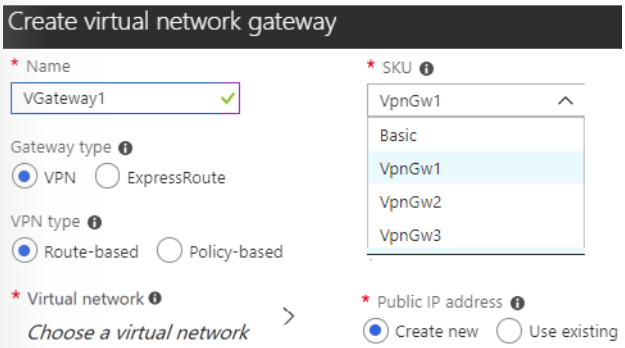
Create the gateway subnet - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal#gatewaysubnet**[24]

VPN Gateway



A VPN gateway is a specific type of virtual network gateway that is used to send encrypted traffic between an Azure virtual network and an on-premises location over the public Internet. You can also use a VPN gateway to send encrypted traffic between Azure virtual networks over the Microsoft network. Each virtual network can have only one VPN gateway. However, you can create multiple connections to the same VPN gateway. When you create multiple connections to the same VPN gateway, all VPN tunnels share the available gateway bandwidth.

24 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal



- **Name and Gateway Type**. Name your gateway and use the VPN Gateway type.
- **VPN Type**. Most VPN types are Route-based.
- **SKU**. Use the drop-down to select a **gateway SKU**[25]. Your choice will affect the number of tunnels you can have and the aggregate throughput benchmark. The benchmark is based on measurements of multiple tunnels aggregated through a single gateway. It is not a guaranteed throughput due to Internet traffic conditions and your application behaviors.
- **Virtual Networks**. Associate a virtual network with the gateway. Before you do this, you must configure the Gateway subnet. Each virtual network will need its own VPN gateway.
- **Public IP Address**. The gateway needs a public IP address to enable it to communicate with the remote network. Make a note of this information. You will need the address when you configure your VPN device.

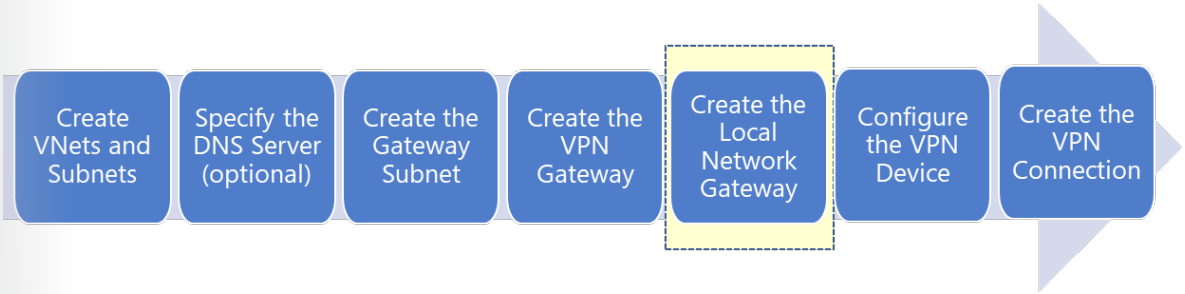
It can take up to 45 minutes to provision the VPN gateway.

✓ After the gateway is created, view the IP address that has been assigned to it by looking at the virtual network in the portal. The gateway should appear as a connected device. In this last step you will create a connection for the device.

For more information, you can see:

Create the VPN gateway - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-how-to-site-to-site-resource-manager-portal#VNetGateway**[26]

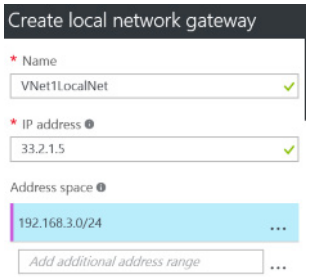
Local Network Gateway



The local network gateway typically refers to the on-premises location. You give the site a name by which Azure can refer to it, then specify the IP address of the on-premises VPN device for the connection. You

25 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpn-gateway-settings
26 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal

also specify the IP address prefixes that will be routed through the VPN gateway to the VPN device. The address prefixes you specify are the prefixes located in the on-premises network.



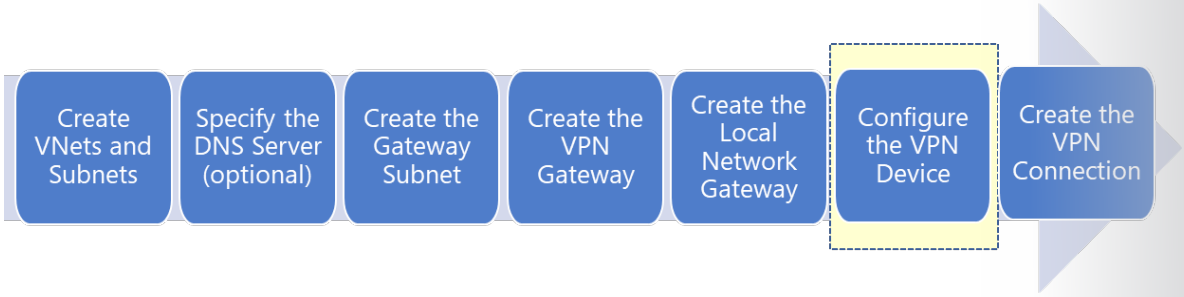
IP Address. The public IP address of the local gateway.

Address Space. One or more IP address ranges (in CIDR notation) that define your local network's address space. For example: 192.168.0.0/16. If you plan to use this local network gateway in a BGP-enabled connection, then the minimum prefix you need to declare is the host address of your BGP Peer IP address on your VPN device.

For more information, you can see:

Create the local network gateway - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal#LocalNetworkGateway**[27]

Configure the VPN device



Microsoft has validated a list of standard VPN devices that should work well with the VPN gateway. This list was created in partnership with device manufacturers like Cisco, Juniper, Ubiquiti, and Barracuda Networks. If you don't see your device listed in the validated VPN devices table (reference link), your device may still work with a Site-to-Site connection. Contact your device manufacturer for additional support and configuration instructions.

To configure your VPN device, you need the following:

- **A shared key**. This is the same shared key that you will specify when creating the VPN connection (next step).
- **The public IP address of your VPN gateway**. When you created the VPN gateway you may have configured a new public IP address or used an existing IP address.

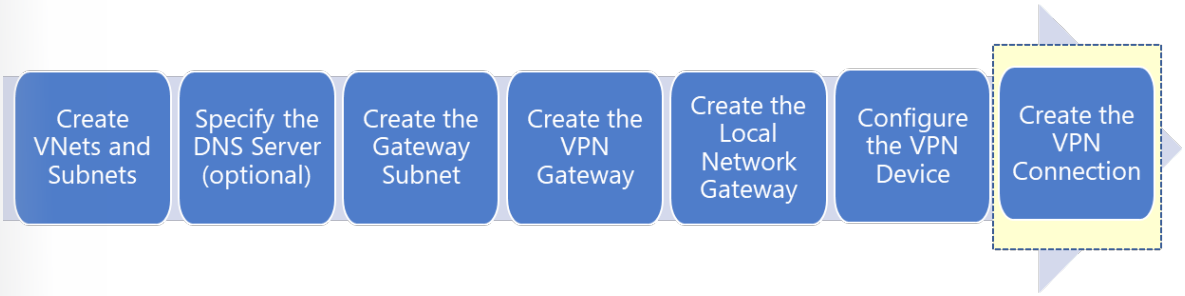
✓ Depending on the VPN device that you have, you may be able to **download a VPN device configuration script**[28].

27 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal
28 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-download-vpndevicescript

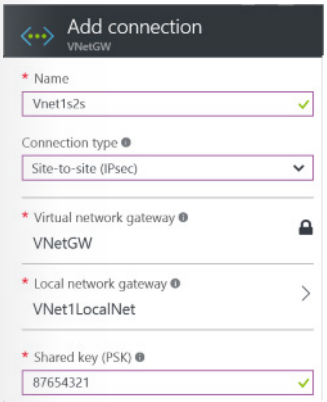
For more information, you can see:

Validated VPN devices list - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpn-devices#devicetable**[29]

Configure the VPN Connection

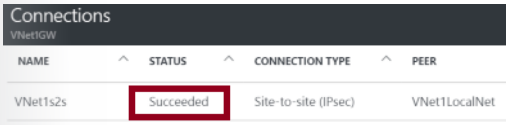


In this step you will configure the connection between the Azure VPN gateway and the local network gateway.



For **Shared Key**, the value here must match the value that you are using for your local VPN device. If your VPN device on your local network doesn't provide a shared key, you can make one up and input it here and on your local device. The important thing is that the shared keys match.

When the connection is complete, you'll see it appear in the Connections blade for your Gateway.



For more information, you can see:

Create the VPN connection - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal#CreateConnection**[30]

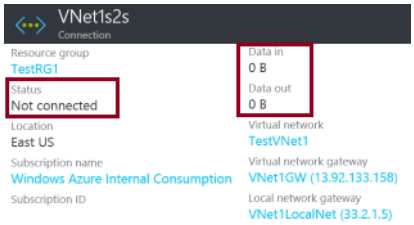
29 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpn-devices
30 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal

Verify the VPN Connection

After you have configured all the Site-to-Site components it is time to verify that everything is working. You can verify the connections either in the portal, or by using PowerShell.

Portal

When you view your connection in the Azure portal the Status should be Succeeded or Connected. Also, you should have data flowing in the Data in and Data out information.



PowerShell

To verify your connection with PowerShell, use the Get-AzureRmVirtualNetworkGatewayConnection cmdlet. For example,

```
Get-AzureRmVirtualNetworkGatewayConnection -Name MyGWConnection -Resource-
GroupName MyRG
```

After the cmdlet has finished, view the values. The connection status should show 'Connected' and you can see ingress and egress bytes.

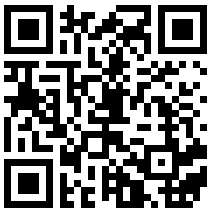
```
"connectionStatus": "Connected",
"ingressBytesTransferred": 33509044,
"egressBytesTransferred": 4142431
```

✓ There is a practice (coming up) that includes all the PowerShell equivalent commands to configure the Site-to-Site connection. Be sure to try it.

For more information, you can see:

Verify the VPN connection - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal#VerifyConnection**[31]

Demonstration: Site-to-Site VPN



31 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal

Practice: Site-to-Site VPN Connections

In this lesson we stepped through the **Create a Site-to-Site connection in the Azure**[32] portal how-to guide. Take some time to explore the portal and make sure you can identity each area that was covered.

There are several QuickStart Templates that might be of interest to you. For example, **Create a Site-to-Site VPN Connection**[33]. As you have time, experiment with the templates.

✓ If you prefer, use the reference links to try this practice with PowerShell and the CLI.

For more information, you can see:

Create a VNet with a Site-to-Site VPN connection using PowerShell - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-create-site-to-site-rm-powershell**

Create a virtual network with a Site-to-Site VPN connection using CLI - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-cli**

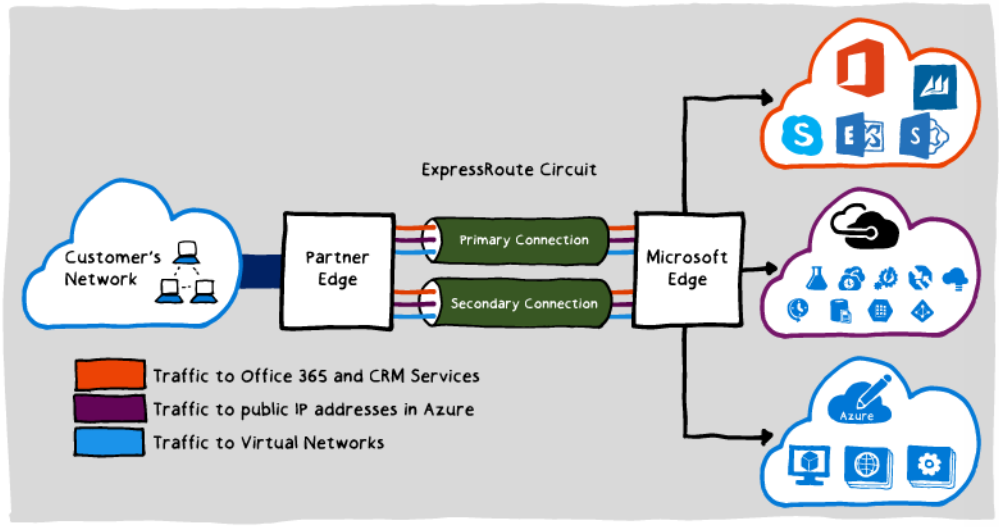
Azure QuickStart Templates - **https://azure.microsoft.com/en-us/resources/templates/**

32 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal
33 https://azure.microsoft.com/en-us/resources/templates/201-site-to-site-vpn/

ExpressRoute

ExpressRoute

Microsoft Azure ExpressRoute lets you extend your on-premises networks into the Microsoft cloud over a dedicated private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure, Office 365, and CRM Online.



ExpressRoute connections do not go over the public Internet. This allows ExpressRoute connections to offer more reliability, faster speeds, lower latencies, and higher security than typical connections over the Internet.

ExpressRoute is excellent for scenarios like periodic data migration, replication for business continuity, disaster recovery, and other high-availability strategies. It can be a cost-effective option for transferring large amounts of data, such as datasets for high-performance computing applications, or moving large virtual machines between your dev-test environment in an Azure virtual private cloud and your on-premises production environment.

You can also use ExpressRoute to add compute and storage capacity to your existing datacenter. With high throughput and fast latencies, Azure will feel like a natural extension to your datacenter, so you enjoy the scale and economics of the public cloud without having to compromise on network performance.

For more information, you can see:

ExpressRoute - **https://azure.microsoft.com/en-us/services/expressroute/**

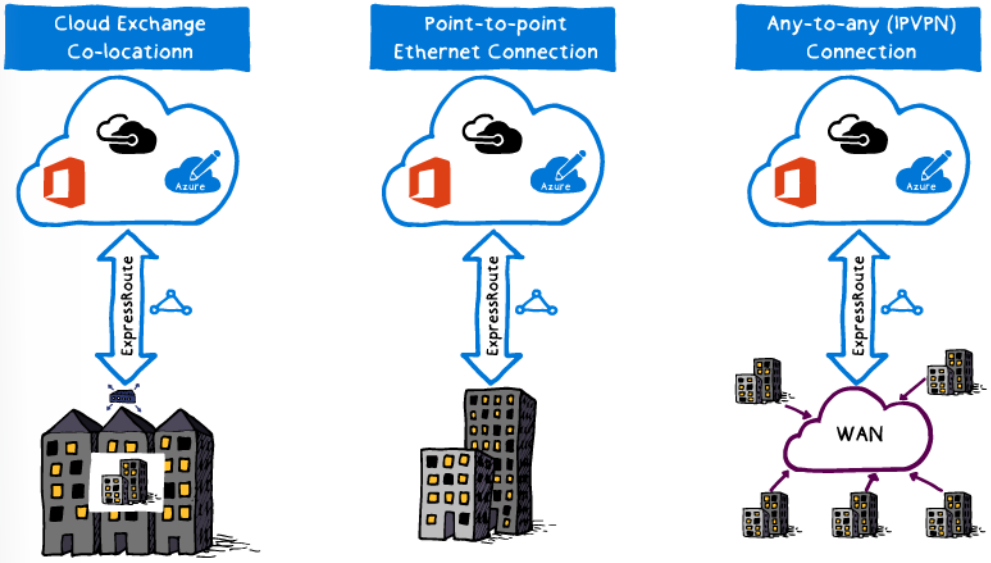
ExpressRoute pricing - **https://azure.microsoft.com/en-us/pricing/details/expressroute/**

ExpressRoute Connection Options

You can create a connection between your on-premises network and the Microsoft cloud in three different ways, **CloudExchange Co-location**[34], **Point-to-point Ethernet Connection**[35], and **Any-to-any**

34 https://docs.microsoft.com/en-us/azure/expressroute/expressroute-connectivity-models
35 https://docs.microsoft.com/en-us/azure/expressroute/expressroute-connectivity-models

(IPVPN) Connection[36]. Connectivity providers can offer one or more connectivity models. You can work with your connectivity provider to pick the model that works best for you.



CloudExchange Co-location

If you are co-located in a facility with a cloud exchange, you can order virtual cross-connections to the Microsoft cloud through the co-location provider's Ethernet exchange. Co-location providers can offer either Layer 2 cross-connections, or managed Layer 3 cross-connections between your infrastructure in the co-location facility and the Microsoft cloud.

Point-to-point Ethernet connections

You can connect your on-premises datacenters/offices to the Microsoft cloud through point-to-point Ethernet links. Point-to-point Ethernet providers can offer Layer 2 connections, or managed Layer 3 connections between your site and the Microsoft cloud.

Any-to-any (IPVPN) networks

You can integrate your WAN with the Microsoft cloud. IPVPN providers, typically Multiprotocol Label Switching (MPLS) VPN, offer any-to-any connectivity between your branch offices and datacenters. The Microsoft cloud can be interconnected to your WAN to make it look just like any other branch office. WAN providers typically offer managed Layer 3 connectivity.

✓ ExpressRoute capabilities and features are all identical across all the above connectivity models.

For more information, you can see:

ExpressRoute connectivity models - **https://docs.microsoft.com/en-us/azure/expressroute/express-route-connectivity-models**

36 https://docs.microsoft.com/en-us/azure/expressroute/expressroute-connectivity-models

Demonstration: ExpressRoute Circuits

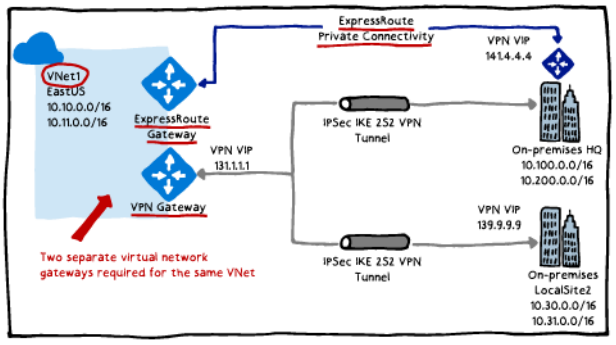
Video: ExpressRoute Performance Monitoring

Site-to-Site and ExpressRoute Coexisting Connections

ExpressRoute is a direct, private connection from your WAN (not over the public Internet) to Microsoft Services, including Azure. Site-to-Site VPN traffic travels encrypted over the public Internet. Being able to configure Site-to-Site VPN and ExpressRoute connections for the same virtual network has several advantages.

You can configure a Site-to-Site VPN as a secure failover path for ExpressRoute or use Site-to-Site VPNs to connect to sites that are not part of your network, but that are connected through ExpressRoute. Notice that this configuration requires two virtual network gateways for the same virtual network, one using the gateway type 'VPN', and the other using the gateway type 'ExpressRoute'.

ExpressRoute and VPN Gateway coexisting connections example



✓ Currently, the deployment options for S2S and ExpressRoute coexisting connections are only possible through PowerShell, and not the Azure portal.

For more information, see:

Site-to-Site and ExpressRoute coexisting connections – **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways#coexisting**[37]

Practice: ExpressRoute and Site-to-Site Coexistence

You can configure ExpressRoute and Site-to-Site VPN connections to work together to take advantage of a couple of private connectivity scenarios. First, this configuration will allow you to use Site-to-Site VPN as a secure failover path for your ExpressRoute connection. Alternatively, you could use your Site-to-Site VPN connection to connect to sites that not connected through ExpressRoute.

The steps in **this practice**[38] covers both scenarios with two different options, depending on whether you already have an existing VNET or need to create one.

✓ The practice procedures are based on the Resource Manager deployment model, and use PowerShell. (The configuration is not available in the Azure portal.)

In this practice, you will learn to:

- Create a new virtual network and coexisting connections
- Configure coexisting connections for an already existing VNet
- Add point-to-site configuration to the VPN gateway

✓ Be sure to read through the documentation, even if you decide not to try this configuration. Be aware of the limits and limitations, such as the SKU and types of routing supported.

For more information, you can see:

Site-to-Site and ExpressRoute coexisting connections - **https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways#coexisting**

Configure ExpressRoute and Site-to-Site coexisting connections - **https://docs.microsoft.com/en-us/azure/expressroute/expressroute-howto-coexist-resource-manager**

ExpressRoute FAQ – **https://docs.microsoft.com/en-us/azure/expressroute/expressroute-faqs**

37 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways
38 https://docs.microsoft.com/en-us/azure/expressroute/expressroute-howto-coexist-resource-manager

Online Lab - Configuring and Managing Virtual Networks

Lab Steps

Online Lab - Configuring and Managing Virtual Networks

Topic: Configuring VNet peering and service chaining

Scenario

ADatum Corporation wants to implement service chaining between Azure virtual networks in its Azure subscription.

Objectives

After completing this lab, you will be able to:

- Deploy Azure VMs by using Azure Resource Manager templates.
- Configure VNet peering.
- Implement routing
- Validate service chaining

Lab Setup

Estimated Time: 45 minutes

User Name: **Student**

Password: **Pa55w.rd**

Exercise 1: Creating an Azure lab environment by using deployment templates

The main tasks for this exercise are as follows:

1. Create the first Azure virtual network environment by using an Azure Resource Manager template
2. Create the second Azure virtual network environment by using an Azure Resource Manager template

Task 1: Create the first Azure virtual network environment by using an Azure Resource Manager template

1. From the lab virtual machine, start Microsoft Edge and browse to the Azure portal at **http://portal.azure.com** and sign in by using the Microsoft account that has the Owner role in the target Azure subscription.
2. In the Azure portal, in the Microsoft Edge window, start a **Bash** session within the **Cloud Shell**.

3. If you are presented with the **You have no storage mounted** message, configure storage using the following settings:
   - Subsciption: the name of the target Azure subscription
   - Cloud Shell region: the name of the Azure region that is available in your subscription and which is closest to the lab location
   - Resource group: the name of a new resource group **az3000400-LabRG**
   - Storage account: a name of a new storage account
   - File share: a name of a new file share
4. From the Cloud Shell pane, create two resource groups by running (replace the `<Azure region>` placeholder with the name of the Azure region that is available in your subscription and which is closest to the lab location)

```
az group create --resource-group az3000401-LabRG --location <Azure region> az
group create --resource-group az3000402-LabRG --location <Azure region>
```

1. From the Cloud Shell pane, upload the first Azure Resource Manager template **D:\LabFiles\04\azuredeploy0401.json** into the home directory.
2. From the Cloud Shell pane, upload the parameter file **D:\LabFiles\04\azuredeploy04.parameters.json** into the home directory.
3. From the Cloud Shell pane, deploy the two Azure VMs hosting Windows Server 2016 Datacenter into the first virtual network by running:

```
az group deployment create --resource-group az3000401-LabRG --template-file
azuredeploy0401.json --parameters @azuredeploy04.parameters.json
```

Note: Do not wait for the deployment to complete but proceed to the next task.

Task 1: Create the second Azure virtual network environment by using an Azure Resource Manager template

1. From the Cloud Shell pane, upload the second Azure Resource Manager template **F:\AZ300\Labs\04\azuredeploy0402.json** into the home directory.
2. From the Cloud Shell pane, deploy an Azure VM hosting Windows Server 2016 Datacenter into the second virtual network by running:

```
az group deployment create --resource-group az3000402-LabRG --template-file
azuredeploy0402.json --parameters @azuredeploy04.parameters.json
```

Note: The second template uses the same parameter file.

Note: Do not wait for the deployment to complete but proceed to the next exercise.

Result: After completing this exercise, you should have created two Azure virtual networks hosting Azure VMs running Windows Server 2016 Datacenter.

Exercise 2: Configuring VNet peering

The main tasks for this exercise are as follows:

1. Configure VNet peering for the first virtual network
2. Configure VNet peering for the second virtual network

Task 1: Configure VNet peering for the first virtual network

1. In the Microsoft Edge window displaying the Azure portal, navigate to the **az3000401-vnet** virtual network blade.
2. From the **az3000401-vnet** blade, create a VNet peering with the following settings:
   - Name: **az3000401-vnet-to-az3000402-vnet**
   - Virtual network deployment model: **Resource manager**
   - Subscription: the name of the Azure subscription you are using for this lab
   - Virtual network: **az3000402-vnet**
   - Allow virtual network access: **Enabled**
   - Allow forwarded traffic: disabled
   - Allow gateway transit: disabled
   - Use remote gateways: disabled

Task 2: Configure VNet peering for the second virtual network

1. In Microsoft Edge, navigate to the **az3000402-vnet** virtual network blade.
2. From the **az3000402-vnet** blade, create a VNet peering with the following settings:
   - Name: **az3000402-vnet-to-az3000401-vnet**
   - Virtual network deployment model: **Resource manager**
   - Subscription: the name of the Azure subscription you are using for this lab
   - Virtual network: **az3000401-vnet**
   - Allow virtual network access: **Enabled**
   - Allow forwarded traffic: disabled
   - Allow gateway transit: disabled
   - Use remote gateways: disabled

Result: After completing this exercise, you should have configured VNet peering between two virtual networks.

Exercise 3: Implementing routing

The main tasks for this exercise are as follows:

1. Enable IP forwarding

2. Configure user defined routing
3. Configure routing on an Azure VM running Windows Server 2016

Task 1: Enable IP forwarding

1. In Microsoft Edge, navigate to the **az3000401-nic2** blade (the NIC of **az3000401-vm2**)
2. On the **az3000401-nic2** blade, modify the **IP configurations** by setting **IP forwarding** to **Enabled**.

Task 2: Configure user defined routing

1. In the Azure portal, create a new route table with the following settings:
   - Name: **az3000402-rt1**
   - Subscription: the name of the Azure subscription you use for this lab
   - Resource group: **az3000402-LabRG**
   - Location: the same Azure region in which you created the virtual networks
   - BGP route propagation: **Disabled**
2. In the Azure portal, add to the route table a route with the following settings:
   - Route name: **custom-route-to-az3000401-vnet**
   - Address prefix: **10.0.0.0/22**
   - Next hop type: **Virtual appliance**
   - Next hop address: **10.0.1.4**
3. In the Azure portal, associate the route table with the **subnet-1** of the **az3000402-vnet**.

Task 3: Configure routing on an Azure VM running Windows Server 2016

1. On MIA-CL1, from the Azure portal, start a Remote Desktop session to **az3000401-vm2** Azure VM.
2. When prompted to authenticate, specify the following credentials:
   - User name: **Student**
   - Password: **Pa55w.rd1234**
3. Once you are connected to az3000401-vm2 via the Remote Desktop session, from **Server Manager**, install the **Remote Access** server role with the **Routing** role service and all required features.
4. In the Remote Desktop session to az3000401-vm2, start the **Routing and Remote Access** console.
5. In the **Routing and Remote Access** console, run **Routing and Remote Access Server Setup Wizard** and enable **LAN routing**.
6. Start **Routing and Remote Access** service.
7. In the Remote Desktop session to az3000401-vm2, start the **Windows Firewall with Advanced Security** console and enable **File and Printer Sharing (Echo Request - ICMPv4-In)** inbound rule for all profiles.

Result: After completing this exercise, you should have configured custom routing within the second virtual network.

Exercise 4: Validating service chaining

The main tasks for this exercise are as follows:

1. Configure Windows Firewall with Advanced Security on an Azure VM
2. Test service chaining between peered virtual networks

Task 1: Configure Windows Firewall with Advanced Security on the target Azure VM

1. On MIA-CL1, from the Azure portal, start a Remote Desktop session to **az3000401-vm1** Azure VM.
2. When prompted to authenticate, specify the following credentials:
   - User name: **Student**
   - Password: **Pa55w.rd1234**
3. In the **Remote Desktop** session to **az3000401-vm1**, start the Windows Firewall with **Advanced Security** console and enable **File and Printer Sharing** (Echo Request - ICMPv4-In) inbound rule for all profiles..

Task 2: Test service chaining between peered virtual networks

1. On MIA-CL1, from the Azure portal, start a Remote Desktop session to **az3000402-vm1** Azure VM.
2. When prompted to authenticate, specify the following credentials:
   - User name: **Student**
   - Password: **Pa55w.rd1234**
3. Once you are connected to az3000402-vm1 via the Remote Desktop session, start **Windows PowerShell**.
4. In the **Windows PowerShell** window, run the following:

```
Test-NetConnection -ComputerName 10.0.0.4 -TraceRoute
```

1. Verify that the test is successful and note that the connection was routed over 10.0.1.4

Result: After completing this exercise, you should have validated service chaining between peered virtual networks.

Review Questions

Module 3 Review Questions

Load Balancer SKUs

You manage an application which uses SQL Server for data storage. Instances of the application that connect to SQL Server vary with respect to their compute needs.

You need ensure the SQL Server instance is load-balanced to optimize performance.

What load balancing options are available? What should you consider before you implement load balancing?

Suggested Answer ↓

When you create an Azure Load Balancer you will select for the type (Internal or Public) of load balancer. You will also select the SKU. The load balancer supports both Basic and Standard SKUs, each differing in scenario scale, features, and pricing. The Standard Load Balancer is the newer Load Balancer product with an expanded and more granular feature set over Basic Load Balancer. It is a superset of Basic Load Balancer.

Load Balancer SKUs

You manage an online training platform that provides consumers with online videos and text-based content. Videos files are often very large.

You must place video content geographically close to customers.

You need to recommend a load-balancing solution that redirects user video requests to the closest CDN.

Which should you recommend and why?

Suggested Answer ↓

Implement a solution that uses Azure Traffic Manager: It provides DNS-based routing to redirect end user traffic to globally distributed end points.

On Demand Capacity

A company is expanding rapidly to new geographical locations. You need to ensure that the company can quickly provide infrastructure and services for new remote offices.

How can you provide access to on-premises services from Azure? What other scenarios should you consider?

Suggested Answer ↓

There are many scenarios where Site-to-Site connections can be useful. Here are a few.

- **Capacity On-Demand**

Azure provides capacity on demand. By creating a connection to Azure, more storage or compute resources can easily be brought online.

- **Strategic Migration**

There are many strategic reasons for moving to Azure. Organizations whose core purpose is not related

to managing complex datacenter deployments, may want to shed competing interests and focus on improving their core business. They may also want to reduce costs by moving to a pay as you go model.

• **Disaster Recovery**

The cloud offers an efficient, cost effective choice for data backup and recovery. Most cloud platforms let you run third-party software for backup and disaster recovery, but with Microsoft these services are fully integrated and easy to turn on, which means you do not have to install and manage a separate product in the cloud.

Module 4 Module Determining Azure Workload Requirements

Overview of Customer Case Study

Customer Situation

Contoso is a US-based financial company based in Boston. There are three additional local branches across the United States. The main datacenter is connected to the internet with a fiber metro Ethernet connection (500 Mbps). Each branch is connected locally to the internet using business class connections, with IPSec VPN tunnels back to the main datacenter. This allows the entire network to be permanently connected, and optimizes internet connectivity.

Contoso has one main datacenter in its primary location. The main datacenter is fully virtualized with VMware. Contoso has 100 ESXi 6.5 virtualization hosts, managed by vCenter Server 6.5.

Contoso uses Active Directory for identity management, and DNS servers on the internal network. The domain controllers in the datacenter run on VMware VMs. The domain controllers at local branches run on physical servers.

Contoso plans to migrate majority of its workloads to Azure. Contoso's IT leadership team has worked closely with the company's business partners to understand what the business wants to achieve with this migration:

- Address business growth: Contoso is growing. As a result, pressure has increased on the company's on-premises systems and infrastructure.
- Increase efficiency: Contoso needs to remove unnecessary procedures and streamline processes for its developers and users. The business needs IT to be fast and to not waste time or money, so the company can deliver faster on customer requirements.

- Increase agility: Contoso IT needs to be more responsive to the needs of the business. It must be able to react faster than the changes that occur in the marketplace for the company to be successful in a global economy. IT at Contoso must not get in the way or become a business blocker.
- Scale: As the company's business grows successfully, Contoso IT must provide systems that can grow at the same pace.

As Contoso considers migrating to Azure, the company wants to run a technical and financial assessment to determine whether its on-premises workloads are suitable for migration to the cloud. In particular, the Contoso team wants to assess machine and database compatibility for migration. It wants to estimate capacity and costs for running Contoso's resources in Azure. Contoso team is also interested in leveraging the company's Software Assurance contract in order to minimize costs when running migrated workloads in Azure.

To get started and to better understand the technologies involved, Contoso plans to assess two of its on-premises apps, summarized in the following table. The company will assess migration scenarios that rehost and refactor the apps.

App name	Platform	App tiers
SmartHotel360 (manages Contoso travel requirements)	Windows Server 2008 R2 with a SQL Server 2008 R2 database	Two-tiered app. The front-end ASP.NET website runs on one VM (WEBVM) and the SQL Server runs on another VM (SQLVM).
osTicket (Contoso service desk app, tracking issues for internal employees and external customers)	Linux Ubuntu 16.04 LTS, Apache 2, and MySQL 5.7 with MySQL PHP 7.0 (LAMP)	Two-tiered app. A front-end PHP website runs on one VM (OSTICKETWEB) and the MySQL database runs on another VM (OSTICKETMYSQL).

Assessment Goals

The Contoso cloud team has identified goals for its migration assessments:

1. After migration, apps in Azure should have the same performance capabilities that they have today in Contoso's on-premises VMWare environment. Moving to the cloud does not mean that app performance is less critical.
2. Contoso needs to understand the compatibility of its applications and databases with Azure requirements. Contoso also needs to understand its hosting options in Azure.
3. Contoso's database administration should be minimized after apps move to the cloud. At the same time, Contoso would like to minimize impact of any potential compatibility issues of its SQL Server-based workloads.
4. Contoso wants to understand not only its migration options, but also the costs associated with the infrastructure after it moves to the cloud.

Assessment Tools

Contoso will use Microsoft tools for its migration assessment. The tools align with the company's goals and should provide Contoso with all the information it needs.

Technology	Description	Cost
Data Migration Assistant	Contoso will use Data Migration Assistant to assess and detect compatibility issues that might affect its database functionality in Azure. Data Migration Assistant assesses feature parity between SQL sources and targets. It recommends performance and reliability improvements.	Data Migration Assistant is a free, downloadable tool.
Azure Migrate	Contoso will use the Azure Migrate service to assess its VMware VMs. Azure Migrate assesses the migration suitability of the machines. It provides sizing and cost estimates for running in Azure.	As of May 2018, Azure Migrate is a free service.
Service Map	Azure Migrate will use Service Map to show dependencies between machines that the company wants to migrate.	Service Map is part of Azure Log Analytics. Currently, Contoso can use Service Map for 180 days without incurring charges.

Assessment Architecture

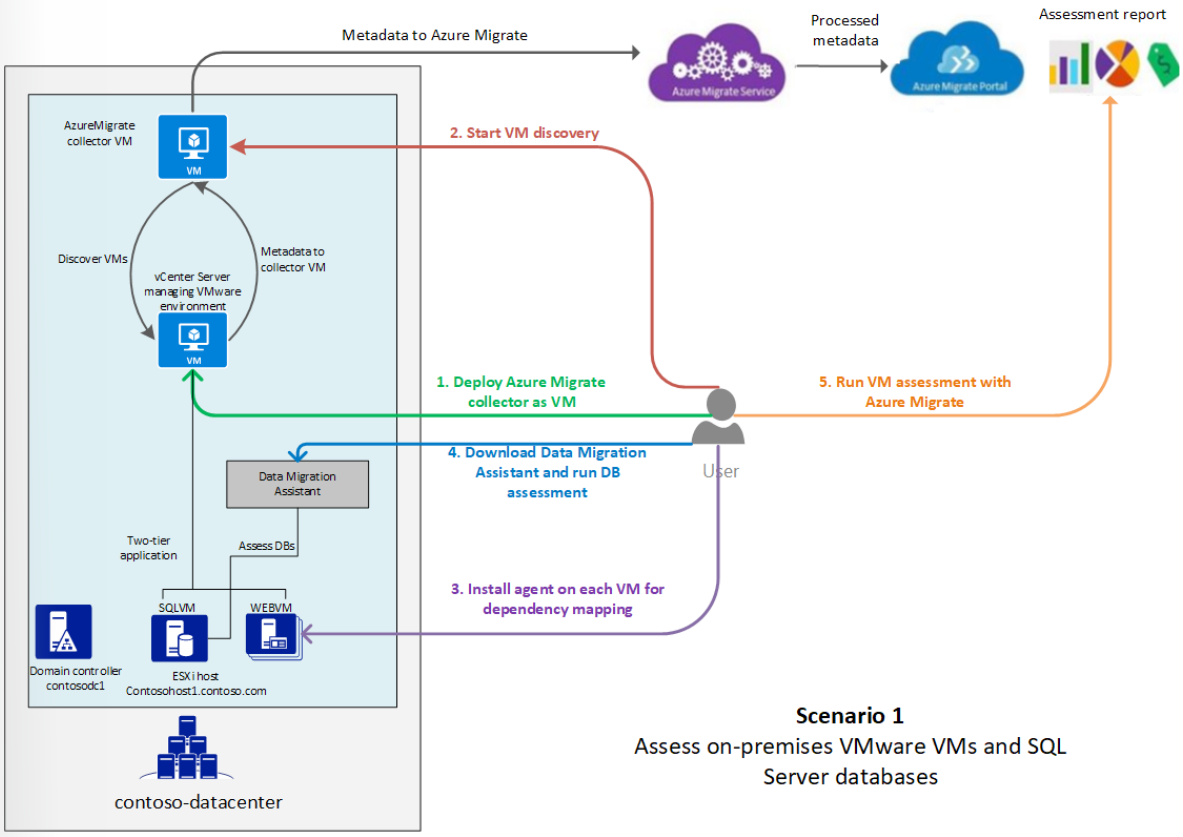
Contoso has an on-premises datacenter (contoso-datacenter) and on-premises domain controllers (CONTOSODC1, CONTOSODC2). VMware ESXi 6.5 hosts include contosohost1 and contosohost2. The VMware environment is managed by vCenter Server 6.5, running on the vcenter.contoso.com VM.

The SmartHotel360 travel app has these characteristics:

- The app is tiered across two VMware VMs (WEBVM and SQLVM).
- The VMs are located on VMware ESXi host contosohost1.contoso.com.
- The VMs are running Windows Server 2008 R2 Datacenter with SP1.

The osTicket service desk app has the following characteristics:

- The app is tiered across two VMs (OSTICKETWEB and OSTICKETMYSQL).
- The VMs are running Ubuntu Linux Server 16.04-LTS.
- OSTICKETWEB is running Apache 2 and PHP 7.0.
- OSTICKETMYSQL is running MySQL 5.7.22.



Scenario 1
Assess on-premises VMware VMs and SQL Server databases

Prerequisites

Contoso must ensure that the following requirements are in place in order to perform an assessment:

- An Azure subscription and either a Microsoft account (MSA) or "Work or School" account with the Owner or Contributor role in the subscription.
- An on-premises vCenter Server instance running version 6.5, 6.0, or 5.5.
- A read-only account in vCenter Server, or permissions to create one.
- Permissions to create a VM on the vCenter Server instance by using an .ova template.
- At least one ESXi host running version 5.0 or later.
- At least two on-premises VMware VMs, one running a SQL Server database.
- Permissions to install Azure Migrate agents on each ESXi VM.
- Direct connectivity from ESXi VMs to internet. If ESXi VMs do not have direct internet connectivity, Contoso will need to deploy Azure Log Analytics Gateway and redirect agent traffic through it.
- Connectivity to the SQL Server instance running on the ESXi VM, for database assessment.

Assessment Overview

The assessment consists of the following steps:

- Downloading and installing Data Migration Assistant: a Contoso IT technician prepares Data Migration Assistant for assessment of the on-premises SQL Server database.
- Generating the database assessment by using Data Migration Assistant: the Contoso IT technician runs the database assessment.
- Reviewing the database assessment by using Data Migration Assistant: the Contoso IT technician reviews the database assessment.
- Preparing for VM assessment by using Azure Migrate: the Contoso IT technician sets up on-premises accounts and adjusts VMware settings.
- Discovering on-premises VMs by using Azure Migrate: the Contoso IT technician creates an Azure Migrate collector VM. Then, the Contoso IT technician runs the collector to discover VMs for assessment.
- Preparing for dependency analysis by using Azure Migrate: the Contoso IT technician installs Azure Migrate agents on the VMs, so the company can see dependency mapping between VMs.
- Reviewing the VM assessment by using Azure Migrate: the Contoso IT technician checks dependencies, groups the VMs, and runs the assessment. When the assessment is ready, the Contoso IT technician analyzes the assessment in preparation for migration.

Primary References

Migration Strategies

Strategies for migration to the cloud fall into four broad categories: rehost, refactor, rearchitect, or rebuild. The strategy you adopt depends upon your business drivers, and migration goals. You might adopt multiple strategies. For example, you could choose to rehost (lift-and-shift) simple apps, or apps that are not critical to your business, but rearchitect those that are more complex and business-critical.

Strategy	Definition	When to use
Rehost	Often referred to as a "lift-and-shift" migration. This option doesn't require code changes, and lets you migrate your existing apps to Azure quickly. Each app is migrated as is, to reap the benefits of the cloud, without the risk and cost associated with code changes.	When you need to move apps quickly to the cloud. When you want to move an app without modifying it. When your apps are architected so that they can leverage Azure IaaS scalability after migration. When apps are important to your business, but you don't need immediate changes to app capabilities.
Refactor	Often referred to as "repackaging," refactoring requires minimal changes to apps, so that they can connect to Azure PaaS, and use cloud offerings. For example, you could migrate existing apps to Azure App Service or Azure Kubernetes Service (AKS). Or, you could refactor relational and non-relational databases into options such as Azure SQL Database Managed Instance, Azure Database for MySQL, Azure Database for PostgreSQL, and Azure Cosmos DB.	If your app can easily be repackaged to work in Azure. If you want to apply innovative DevOps practices provided by Azure, or you are thinking about DevOps using a container strategy for workloads. For refactoring, you need to think about the portability of your existing code base, and available development skills.
Rearchitect	Rearchitecting for migration focuses on modifying and extending app functionality and the code base to optimize the app architecture for cloud scalability. For example, you could break down a monolithic application into a group of microservices that work together and scale easily. Or, you could rearchitect relational and non-relational databases to a fully managed DBaaS solutions, such as Azure SQL Database Managed Instance, Azure Database for MySQL, Azure Database for PostgreSQL, and Azure Cosmos DB.	When your apps need major revisions to incorporate new capabilities, or to work effectively on a cloud platform. When you want to use existing application investments, meet scalability requirements, apply innovative Azure DevOps practices, and minimize use of virtual machines.
Rebuild	Rebuild takes things a step further by rebuilding an app from scratch using Azure cloud technologies. For example, you could build green field apps with cloud-native technologies like Azure Functions, Azure AI, Azure SQL Database Managed Instance, and Azure Cosmos DB.	When you want rapid development, and existing apps have limited functionality and lifespan. When you're ready to expedite business innovation (including DevOps practices provided by Azure), build new applications using cloud-native technologies, and take advantage of advancements in AI, Blockchain, and IoT.

Step-by-Step: Determining Azure Workload Requirements

Step 1: Downloading and Installing Data Migration Assistant

- The Contoso IT technician downloads Data Migration Assistant from the Microsoft Download Center. Data Migration Assistant can be installed on any machine with direct connectivity to the SQL Server instance. Contoso Data Migration Assistant should not be run on the SQL Server host machine.
- The Contoso IT technician runs the downloaded setup file (DownloadMigrationAssistant.msi) to begin the installation.
- On the Finish page, the Contoso IT technician selects Launch Microsoft Data Migration Assistant before finishing the wizard.

Get Data Migration Assistant[1]

To install DMA, download the latest version of the tool from the **Microsoft Download Center**[2], and then run the **DataMigrationAssistant.msi** file.

Capabilities[3]

- Assess on-premises SQL Server instance(s) migrating to Azure SQL database(s). The assessment workflow helps you to detect the following issues that can affect Azure SQL database migration and provides detailed guidance on how to resolve them.
  - Migration blocking issues: Discovers the compatibility issues that block migrating on-premises SQL Server database(s) to Azure SQL Database(s). DMA provides recommendations to help you address those issues.
  - Partially supported or unsupported features: Detects partially supported or unsupported features that are currently in use on the source SQL Server instance. DMA provides a comprehensive set of recommendations, alternative approaches available in Azure, and mitigating steps so that you can incorporate them into your migration projects.
- Discover issues that can affect an upgrade to an on-premises SQL Server. These are described as compatibility issues and are organized in the following categories:
  - Breaking changes
  - Behavior changes
  - Deprecated features
- Discover new features in the target SQL Server platform that the database can benefit from after an upgrade. These are described as feature recommendations and are organized in the following categories:
  - Performance
  - Security

1 https://docs.microsoft.com/en-us/sql/dma/dma-overview?view=sql-server-2017#get-data-migration-assistant
2 https://www.microsoft.com/download/details.aspx?id=53595
3 https://docs.microsoft.com/en-us/sql/dma/dma-overview?view=sql-server-2017#capabilities

  - Storage
- Migrate an on-premises SQL Server instance to a modern SQL Server instance hosted on-premises or on an Azure virtual machine (VM) that is accessible from your on-premises network. The Azure VM can be accessed using VPN or other technologies. The migration workflow helps you to migrate the following components:
  - Schema of databases
  - Data and users
  - Server roles
  - SQL Server and Windows logins
- After a successful migration, applications can connect to the target SQL Server databases seamlessly.

Prerequisites[4]

To run an assessment, you have to be a member of the SQL Server **sysadmin** role.

Supported source and target versions[5]

DMA replaces all previous versions of SQL Server Upgrade Advisor and should be used for upgrades for most SQL Server versions. Supported source and target versions are:

Sources

- SQL Server 2005
- SQL Server 2008
- SQL Server 2008 R2
- SQL Server 2012
- SQL Server 2014
- SQL Server 2016
- SQL Server 2017 on Windows

Targets

- SQL Server 2012
- SQL Server 2014
- SQL Server 2016
- SQL Server 2017 on Windows and Linux
- Azure SQL Database
- Azure SQL Database Managed Instance

4 https://docs.microsoft.com/en-us/sql/dma/dma-overview?view=sql-server-2017#prerequisites
5 https://docs.microsoft.com/en-us/sql/dma/dma-overview?view=sql-server-2017#supported-source-and-target-versions

Step 2: Generating the Database Assessment for SmartHotel360

Step 3: Reviewing the database assessment for SmartHotel360

The Contoso IT technician must run assessment of the on-premises SQL Server database for the SmartHotel360 app.

In Data Migration Assistant, the Contoso IT technician selects New > Assessment, and then gives the assessment a project name.

- For Source server type, the Contoso IT technician selects SQL Server on Azure Virtual Machines.

Note: Currently, Data Migration Assistant does not support assessment for migrating to an Azure SQL Database Managed Instance. As a workaround, the Contoso IT technician uses SQL Server on an Azure VM as the supposed target for the assessment.

By selecting Azure SQL Database Managed Instance as target for migrating its on-premises SQL Server database, Contoso considerably minimizes potential for any compatibility issues during migration. Managed Instance allows existing SQL Server customers to lift and shift their on-premises applications to the cloud with minimal application and database changes. At the same time, Managed Instance preserves all PaaS capabilities, such as automatic patching and version updates, automated backups, and high-availability, effectively drastically reducing management overhead and TCO.

- In Select Target Version, the Contoso IT technician selects SQL Server 2017 as the target version. The Contoso IT technician needs to select this version because this is the version used by SQL Database Managed Instance.
- The Contoso IT technician selects reports to help discover information about compatibility and new features:

Compatibility Issues note changes that might break migration or that require a minor adjustment before migration. This report keeps the Contoso IT technician informed about any features currently in use that are deprecated. Issues are organized by compatibility level.

New features' recommendation notes new features in the target SQL Server platform that can be used for the database after migration. New feature recommendations are organized under the headings Performance, Security, and Storage.

- In Connect to a server, the Contoso IT technician enters the name of the VM running the database and credentials to access it. The Contoso IT technician selects Trust server certificate to make sure the VM can access SQL Server. Then, the Contoso IT technician selects Connect.
- In Add source, the Contoso IT technician adds the database it wants to assess, and then selects Next to start the assessment.
- The assessment is created.

Step 4: Preparing for VM assessment by using Azure Migrate

The Contoso IT technician needs to create a VMware account that Azure Migrate can use to automatically discover VMs for assessment, verify rights to create a VM, note the ports that need to be opened, and assign the statistics settings level.

- Setting up a VMware account: VM discovery requires a read-only account in vCenter Server that has the following properties:

User type: At least a read-only user.

Permissions: For the datacenter object, the Propagate to Child Objects checkbox must be selected. For Role, Read-only must be selected.

Details: The user is assigned at the datacenter level, with access to all objects in the datacenter.

To restrict access, the No access role with the Propagate to child objectto the child objects (vSphere hosts, datastores, VMs, and networks) setting must be assigned

- Verifying permissions to create a VM: the Contoso IT technician verifies that it has permissions to create a VM by importing a file in .ova format.
- Verifying ports: the Contoso assessment uses dependency mapping. Dependency mapping requires an agent to be installed on VMs that will be assessed. The agent must be able to connect to Azure from TCP port 443 on each VM.
- Assigning the statistics settings level: before the deployment begins, the Contoso IT technician must set the statistics settings for the vCenter Server to level 3.

Note: After setting the level, the Contoso IT technician must wait at least a day before running the assessment. Otherwise, the assessment might not work as expected. If the level is higher than 3, the assessment works, but:

Performance data for disks and networking is not collected.

For storage, Azure Migrate recommends a standard disk in Azure, with the same size as the on-premises disk.

For networking, for each on-premises network adapter, a network adapter is recommended in Azure.

For compute, Azure Migrate looks at the VM cores and memory size and recommends an Azure VM with the same configuration. If there are multiple eligible Azure VM sizes, the one with the lowest cost is recommended.

To set the level:

In the vSphere Web Client, the Contoso IT technician opens the vCenter Server instance.

The Contoso IT technician selects Manage > Settings > General > Edit.

In Statistics, the Contoso IT technician sets the statistic level settings to Level 3.

Step 5: Discovering on-premises VMs by using Azure Migrate

To discover VMs, the Contoso IT technician creates an Azure Migrate project. The Contoso IT technician downloads and sets up the collector VM. Then, the Contoso IT technician runs the collector to discover its on-premises VMs.

Creating a project

- In the Azure portal, the Contoso IT technician searches for Azure Migrate. Then, the Contoso IT technician creates a project.
- The Contoso IT technician specifies a project name (ContosoMigration) and the Azure subscription, as well as creates a new Azure resource group (ContosoFailoverRG).
- Note:

You can create an Azure Migrate project only in the West Central US or East US region.

You can plan a migration for any target location.

The project location is used only to store the metadata that's gathered from on-premises VMs.

Downloading the collector appliance: Azure Migrate creates an on-premises VM known as the collector appliance. The VM discovers on-premises VMware VMs and sends metadata about the VMs to the Azure Migrate service. To set up the collector appliance, the Contoso IT technician downloads an OVA template, and then imports it to the on-premises vCenter Server instance to create the VM.

- In the Azure Migrate project, the Contoso IT technician selects Getting Started > Discover & Assess > Discover Machines. The Contoso IT technician downloads the OVA template file.
- Contoso copies the project ID and key. The project and ID are required for configuring the collector.

Verifying the collector appliance: before deploying the VM, the Contoso IT technician checks that the OVA file is secure:

- On the machine on which the file was downloaded, the Contoso IT technician opens an administrator Command Prompt window.
- Contoso runs the following command to generate the hash for the OVA file:
- C:\>CertUtil -HashFile <file_location> [Hashing Algorithm]
- For example:
- C:\>CertUtil -HashFile C:\AzureMigrate\AzureMigrate.ova SHA256
- The generated hash should match these settings (version 1.0.9.15):

Algorithm	Hash value
MD5	e9ef16b0c837638c506b5fc0ef75ebfa
SHA1	37b4b1e92b3c6ac2782ff5258450df6686c89864
SHA256	8a86fc17f69b69968eb20a5c4c288c194cdcffb4ee-6568d85ae5ba96835559ba

Creating the collector appliance: now, the Contoso IT technician can import the downloaded file to the vCenter Server instance and provision the collector appliance VM:

- In the vSphere Client console, the Contoso IT technician selects File > Deploy OVF Template.
- In the Deploy OVF Template Wizard, the Contoso IT technician selects Source, and then specifies the location of the OVA file.
- In Name and Location, the Contoso IT technician specifies a display name for the collector VM. Then, it selects the inventory location in which to host the VM. The Contoso IT technician also specifies the host or cluster on which to run the collector appliance.
- In Storage, the Contoso IT technician specifies the storage location. In Disk Format, the Contoso IT technician selects how it wants to provision the storage.
- In Network Mapping, the Contoso IT technician specifies the network in which to connect the collector VM. The network needs internet connectivity to send metadata to Azure.
- Contoso reviews the settings, and then selects Power on after deployment > Finish. A message that confirms successful completion appears when the appliance is created.

Runing the collector to discover VMs: now, the Contoso IT technician runs the collector to discover VMs. Currently, the collector currently supports only English (United States) as the operating system language and collector interface language.

- In the vSphere Client console, the Contoso IT technician selects Open Console. The Contoso IT technician specifies the language, time zone, and password preferences for the collector VM.
- On the desktop, the Contoso IT technician selects the Run collector shortcut.
- In Azure Migrate Collector, the Contoso IT technician selects Set up prerequisites. The Contoso IT technician accepts the license terms and reads the third-party information.

- The collector checks that the VM has internet access, that the time is synced, and that the collector service is running. (The collector service is installed by default on the VM.) the Contoso IT technician also installs VMware PowerCLI.
- Note: it is assumed that the VM has direct access to the internet without using a proxy.
- In Specify vCenter Server details, the Contoso IT technician enters the name (FQDN) or IP address of the vCenter Server instance and the read-only credentials used for discovery.
- Contoso selects a scope for VM discovery. The collector can discover only VMs that are within the specified scope. The scope can be set to a specific folder, datacenter, or cluster. The scope shouldn't contain more than 1,500 VMs.
- In Specify migration project, the Contoso IT technician enters the Azure Migrate project ID and key that were copied from the portal. To get the project ID and key, the Contoso IT technician can go to the project Overview page > Discover Machines.
- In View collection progress, the Contoso IT technician can monitor discovery and check that metadata collected from the VMs is in scope. The collector provides an approximate discovery time.

Verifying VMs in the portal: when collection is finished, the Contoso IT technician checks that the VMs appear in the portal:

- In the Azure Migrate project, the Contoso IT technician selects Manage > Machines. The Contoso IT technician checks that the VMs that it wants to discover are shown.
- Currently, the machines don't have the Azure Migrate agents installed. Contoso must install the agents to view dependencies.

Step 6: Preparing for Dependency Analysis

To view dependencies between VMs that it wants to assess, the Contoso IT technician downloads and installs agents on the app VMs. The Contoso IT technician installs agents on all VMs for its apps, both for Windows and Linux.

Taking a snapshot: to keep a copy of the VMs before modifying them, the Contoso IT technician takes a snapshot before the agents are installed.

Downloading and installing the VM agents

- In Machines, the Contoso IT technician selects the machine. In the Dependencies column, the Contoso IT technician selects Requires installation.
- In the Discover Machines pane, the Contoso IT technician:

Downloads the Microsoft Monitoring Agent (MMA) and Dependency Agent for each Windows VM.

Downloads the MMA and Dependency Agent for each Linux VM.

- The Contoso IT technician copies the workspace ID and key. The Contoso IT technician needs the workspace ID and key when it installs the MMA.

- Installing the agents on Windows VMs

The Contoso IT technician runs the installation on each VM.

- Installing the MMA on Windows VMs

The Contoso IT technician double-clicks the downloaded agent.

In Destination Folder, the Contoso IT technician keeps the default installation folder, and then selects Next.

In Agent Setup Options, the Contoso IT technician selects Connect the agent to Azure Log Analytics > Next.

In Azure Log Analytics, the Contoso IT technician pastes the workspace ID and key that it copied from the portal.

In Ready to Install, the Contoso IT technician installs the MMA.

- Installing the Dependency agent on Windows VMs

The Contoso IT technician double-clicks the downloaded Dependency Agent.

The Contoso IT technician accepts the license terms and waits for the installation to finish.

- Installing the agents on Linux VMs

The Contoso IT technician runs the installation on each VM.

- Install the MMA on Linux VMs

The Contoso IT technician installs the Python ctypes library on each VM by using the following command:

sudo apt-get install python-ctypeslib

The Contoso IT technician must run the command to install the MMA agent as root. To become root, the Contoso IT technician runs the following command, and then enters the root password:

sudo -i

The Contoso IT technician installs the MMA:

The Contoso IT technician enters the workspace ID and key in the command.

Commands are for 64-bit.

The workspace ID and primary key are located in the Log Analytics workspace in the Azure portal. Select Settings, and then select the Connected Sources tab.

Run the following commands to download the Log Analytics agent, validate the checksum, and install and onboard the agent:

wget https://raw.githubusercontent.com/Microsoft/OMS-Agent-for-Linux/master/installer/scripts/onboard_agent.sh && sh onboard_agent.sh -w 6b7fcaff-7efb-4356-ae06-516cacf5e25d -s k7gAMAw5Bk-8pFVUTZKmk2lG4eUciswzWfYLDTxGcD8pcyc4oT8c6ZRgsMy3MmsQSHuSOcmBUsCjoRiG2x9A8Mg==

- Installing the Dependency Agent on Linux VMs

After the MMA is installed, the Contoso IT technician installs the Dependency Agent on the Linux VMs:

The Dependency Agent is installed on Linux computers by using InstallDependencyAgent-Linux64.bin, a shell script that has a self-extracting binary. The Contoso IT technician runs the file by using sh, or it adds execute permissions to the file itself.

The Contoso IT technician installs the Linux Dependency Agent as root:

wget --content-disposition https://aka.ms/dependencyagentlinux -O InstallDependencyAgent-Linux64.bin && sudo sh InstallDependencyAgent-Linux64.bin –s

Step 7: Running and Analyzing the VM Assessment

- The Contoso IT technician can now verify machine dependencies and create a group. Then, it runs the assessment for the group.
- Verifying dependencies and creating a group

To determine which machines to analyze, the Contoso IT technician selects View Dependencies.

For SQLVM, the dependency map shows the following details:

Process groups or processes that have active network connections running on SQLVM during the specified time period (an hour, by default).

Inbound (client) and outbound (server) TCP connections to and from all dependent machines.

Dependent machines that have the Azure Migrate agents installed are shown as separate boxes.

Machines that don't have the agents installed show port and IP address information.

For machines that have the agent installed (WEBVM), the Contoso IT technician selects the machine box to view more information. The information includes the FQDN, operating system, and MAC address.

The Contoso IT technician selects the VMs to add to the group (SQLVM and WEBVM). The Contoso IT technician uses Ctrl+Click to select multiple VMs.

The Contoso IT technician selects Create Group, and then enters a name (smarthotelapp).

Note: to view more granular dependencies, you can expand the time range. You can select a specific duration or select start and end dates.

- Running an assessment

In Groups, the Contoso IT technician opens the group (smarthotelapp), and then selects Create assessment.

To view the assessment, the Contoso IT technician selects Manage > Assessments.

The Contoso IT technician uses the default assessment settings, but you can customize settings.

- Analyzing the VM assessment

An Azure Migrate assessment includes information about the compatibility of on-premises with Azure, suggested right-sizing for Azure VM, and estimated monthly Azure costs.

- Reviewing confidence rating

An assessment has a confidence rating of from 1 star to 5 stars (1 star is the lowest and 5 stars is the highest).

The confidence rating is assigned to an assessment based on the availability of data points that are needed to compute the assessment.

The rating helps you estimate the reliability of the size recommendations that are provided by Azure Migrate.

The confidence rating is useful when you are doing performance-based sizing. Azure Migrate might not have enough data points for utilization-based sizing. For as on-premises sizing, the confidence rating is always 5 stars because Azure Migrate has all the data points it needs to size the VM.

Depending on the percentage of data points available, the confidence rating for the assessment is provided:

Availability of data points	Confidence rating
0%-20%	1 star
21%-40%	2 stars
41%-60%	3 stars
61%-80%	4 stars
81%-100%	5 stars

- Verifying Azure readiness

The assessment report shows the information that's summarized in the table. To show performance-based sizing, Azure Migrate needs the following information. If the information can't be collected, sizing assessment might not be accurate.

Utilization data for CPU and memory.

Read/write IOPS and throughput for each disk attached to the VM.

Network in/out information for each network adapter attached to the VM.

Setting	Indication	Details
Azure VM readiness	Indicates whether the VM is ready for migration.	Possible states: - Ready for Azure - Ready with conditions - Not ready for Azure - Readiness unknown If a VM is not ready, Azure Migrate shows some remediation steps.
Azure VM size	For ready VMs, Azure Migrate provides an Azure VM size recommendation	Sizing recommendation depends on assessment properties: - If you used performance-based sizing, sizing considers the performance history of the VMs. - If you used as on-premises, sizing is based on the on-premises VM size and utilization data is not used.
Suggested tool	Because Azure machines are running the agents, Azure Migrate looks at the processes that are running inside the machine. It identifies whether the machine is a database machine.	
VM information	The report shows settings for the on-premises VM, including operating system, boot type, and disk and storage information.	

- Reviewing monthly cost estimates

This view shows the total compute and storage cost of running the VMs in Azure. It also shows details for each machine.

Cost estimates are calculated by using the size recommendations for a machine.

Estimated monthly costs for compute and storage are aggregated for all VMs in the group.

Step 8: Cleaning up After Assessment

- When the assessment finishes, the Contoso IT technician retains the Azure Migrate appliance to use in future evaluations.
- The Contoso IT technician turns off the VMware VM. The Contoso IT technician will use it again when it evaluates additional VMs.

- The Contoso IT technician keeps the Contoso Migration project in Azure. The project currently is deployed in the ContosoFailoverRG resource group in the East US Azure region.
- The collector VM has a 180-day evaluation license. If this limit expires, the Contoso IT technician will need to download the collector and set it up again.

Checklist of Assessment Goals

Goal: After Migration

After migration, apps in Azure should have the same performance capabilities that apps have today in Contoso's on-premises VMWare environment. Moving to the cloud doesn't mean that app performance is less critical.

Data Migration Assistant provides an assessment that allows Contoso assess the outcome of migrating its SQL Server environment to Azure SQL Database Managed Instance. The process generates a report listing recommendations supported by the target platform that can be used by the database after migration. The performance recommendations include such features as in-Memory OLTP and columnstore indexes.

- Note: As of December 2018, Data Migration Assistant does not support assessment for migrating to an Azure SQL Database Managed Instance. As a workaround, the Contoso can use SQL Server on an Azure VM as the supposed target for the assessment.

Azure SQL Database Managed Instance is available in the vCore-based purchasing model that allows independent scaling of compute and storage resources, facilitating matching on-premises performance levels. It also offers Contoso the choice of the hardware generation:

- Gen 4 - Up to 24 logical CPUs based on Intel E5-2673 v3 (Haswell) 2.4 GHz processors (where vCore is equal to a physical core), 7 GB per core, attached SSD
- Gen 5 - Up to 80 logical CPUs based on Intel E5-2673 v4 (Broadwell) 2.3 GHz processors (where vCore is equal to a hyperthread), 5.5. GB per core, fast eNVM SSD

Azure Migrate assesses performance of the VMware environment. Azure Migrate leverages an on-premises VM known as the collector appliance. The VM discovers on-premises VMware VMs and sends metadata about the VMs to the Azure Migrate service. An Azure Migrate assessment offers information about the suggested right-sizing for Azure VM. Sizing recommendation depends on the assessment approach:

- When using performance-based sizing, sizing considers the performance history of the VMs, including:

Utilization data for CPU and memory.

Read/write IOPS and throughput for each disk attached to the VM.

Network in/out information for each network adapter attached to the VM.

- When using as on-premises, sizing is based on the on-premises VM size and utilization data is not used.

In addition, Contoso installs Microsoft Monitoring Agent (MMA) and Dependency Agent on the app VMs in order to identify application dependencies. Any dependency might need to be considered for inclusion in the scope of migration in order to minimize negative performance impact resulting from increased latency in cross-premises connectivity scenarios.

Goal: Understanding Compatibility

Contoso needs to understand the compatibility of its applications and databases with Azure requirements. Contoso also needs to understand its hosting options in Azure.

In Azure, customers can run SQL Server workloads running in a hosted infrastructure (IaaS) or running as a hosted service (PaaS):

- Azure SQL Database: A SQL database engine, based on the Enterprise Edition of SQL Server that is optimized for modern application development. Azure SQL Database offers several deployment options:

A single database on a logical server.

A database in an elastic pool sharing resources with other databases on the same logical server.

An Azure SQL Database Managed Instances.

With all three versions, Azure SQL Database adds additional features that are not available in SQL Server, such as built-in intelligence and management. A logical server containing single and pooled databases offers most of database-scoped features of SQL Server. With Azure SQL Database Managed Instance, Azure SQL Database offers shared resources for databases and additional instance-scoped features. Azure SQL Database Managed Instance supports database migration with minimal to no database change.

- SQL Server on Azure Virtual Machines: SQL Server installed and hosted in the cloud on Windows Server or Linux virtual machines (VMs) running on Azure, also known as an infrastructure as a service (IaaS). SQL Server on Azure virtual machines is a good option for migrating on-premises SQL Server databases and applications without any database change. All recent versions and editions of SQL Server are available for installation in an IaaS virtual machine. The most significant difference from SQL Database is that SQL Server VMs allow full control over the database engine. You can choose when maintenance/patching will start, to change the recovery model to simple or bulk logged to enable faster load less log, to pause or start engine when needed, and you can fully customize the SQL Server database engine. With this additional control comes with added responsibility to manage the virtual machines.

Due to its requirement to minimize both management overhead and potential migration issues, Contoso wants to assess migration to Azure SQL Database Managed Instance.

Data Migration Assistant provides an assessment that allows Contoso assess the outcome of migrating its SQL Server environment to Azure SQL Database Managed Instance. The process generates a report that identifies compatibility issues affecting database migration. Compatibility Issues note changes that might break migration or that require a minor adjustment before migration. The report keeps Contoso informed about any features currently in use that are deprecated. Issues are organized by compatibility level. Compatibility levels map to SQL Server versions as follows:

- 100: SQL Server 2008/Azure SQL Database
- 110: SQL Server 2012/Azure SQL Database
- 120: SQL Server 2014/Azure SQL Database
- 130: SQL Server 2016/Azure SQL Database

- 140: SQL Server 2017/Azure SQL Database

An Azure Migrate assessment includes information about the compatibility of on-premises with Azure. Azure VM readiness indicates whether the VM is ready for migration. Possible states include:

- Ready for Azure
- Ready with conditions
- Not ready for Azure
- Readiness unknown

If a VM is not ready, Azure Migrate offers typically remediation steps.

Goal: Minimize Impact of Potential Compatibility Issues

Contoso's database administration should be minimized after apps move to the cloud. At the same time, Contoso would like to minimize impact of any potential compatibility issues of its SQL Server-based workloads.

By selecting Azure SQL Database Managed Instance as target for migrating its on-premises SQL Server database, Contoso considerably minimizes potential for any compatibility issues during migration. Managed Instance allows existing SQL Server customers to lift and shift their on-premises applications to the cloud with minimal application and database changes. At the same time, Managed Instance preserves all PaaS capabilities, such as automatic patching and version updates, automated backups, and high-availability, effectively drastically reducing management overhead and TCO.

Goal: Costs Associated with the Infrastructure

Contoso wants to understand not only its migration options, but also the costs associated with the infrastructure after it moves to the cloud.

Azure SQL Database Managed Instance is available in the vCore-based purchasing model that will enable Contoso to choose the exact amount of storage capacity and compute needed for the migrated workload. Details regarding pricing for SQL Server Managed Instance are available at **https://azure.microsoft.com/en-us/pricing/details/sql-database/managed/**

An Azure Migrate assessment includes information about the estimated monthly Azure costs. The estimates include the total compute and storage cost of running the VMs in Azure as well as details for each machine. Cost estimates are calculated by using the size recommendations for a machine. Estimated monthly costs for compute and storage are aggregated for all VMs in the group.

In addition, with Software Assurance, Contoso will be able to leverage their existing Windows Server and SQL Server licenses for discounted rates on Azure VMs and SQL Database Managed Instances using the Azure Hybrid Benefit for Windows Server and SQL Server.