

Introduction to Azure virtual machines

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/intro-to-azure-virtual-machines/1-introduction>

Introduction

Completed

Suppose you work for a company doing medical research and you're responsible for managing the on-premises servers. The servers you administer run all the company infrastructure, from web servers to databases. However, the hardware is aging and starting to struggle to keep up with some of the new data analysis applications being deployed to it.

You could upgrade all the hardware, but that's not appealing for several reasons:

- The servers are physically scattered all around the world with minimal staff in each location. We'd like to centralize the upgrade to our home office.
- The company runs custom data analysis software on several versions and flavors of Windows and Linux, sometimes set up with odd configurations that aren't entirely understood. We need a way to test our deployments completely and try different configurations to make sure everything is working before we transition the work.
- Business is booming, and the company is growing fast. It's likely that the load on the internal servers, particularly the databases, will continue to grow. This growth requires us to either buy for the future or come up with a scaling plan to handle the growth.

For these reasons, you decide that it's time to explore the cloud to see if it can help solve the load and scale problem. Since you have a bunch of mixed servers and custom software, it makes sense to look at trying to move servers one at a time into Azure using Azure Virtual Machines (VMs).

Azure VMs are one of several types of on-demand, scalable computing resources that Azure offers. With VMs, you have total control over the configuration and can install anything you need to perform the work. You don't need to purchase physical hardware when you need to scale or extend your datacenter. Finally, Azure provides other services to monitor, secure, and manage updates and patches to the OS.

We're going to look at the decisions made before creating a VM, the options to create and manage the VM, and the extensions and services you use to manage your VM.

Learning objectives

In this module, you learn how to:

- Compile a checklist for creating a virtual machine
- Describe the options to create and manage virtual machines
- Describe the other services available to administer virtual machines

2. Compile a checklist for creating an Azure Virtual Machine

<https://learn.microsoft.com/en-us/training/modules/intro-to-azure-virtual-machines/2-compile-a-checklist-for-creating-a-vm>

Compile a checklist for creating an Azure Virtual Machine

Completed

Performing a migration of on-premises servers to Azure requires planning and care. You can move them all at once, or more likely, in small batches or even individually. Before you create a single VM, you should sit down and sketch out your current infrastructure model and see how it might map to the cloud.

What is an Azure resource?

An **Azure resource** is a manageable item in Azure. Just like a physical computer in your datacenter, VMs have several elements that are needed to do their job:

- The VM itself
- Disks for storage
- Virtual network
- Network interface to communicate on the network
- Network Security Group (NSG) to secure the network traffic
- An IP address (public, private, or both)

Azure creates all of these resources if necessary, or you can supply existing ones as part of the deployment process. Each resource needs a name that's used to identify it. If Azure creates the

resource, it uses the VM name to generate a resource name - another reason to be consistent with your VM names!

Required resources for IaaS Virtual Machines

Let's walk through a checklist of things to think about.

- The network
- VM name
- Location
- The VM size
- Disks
- Operating system

The network

The first thing you should think about isn't the virtual machine at all - it's the network. Take a look at one of your on-premises servers:

- What does the server communicate with?
- Which ports are open?

Virtual networks (VNETs) are used in Azure to provide private connectivity between Azure Virtual Machines and other Azure services. VMs and services that are part of the same virtual network can access one another. By default, services outside the virtual network can't connect to services within the virtual network. You can, however, configure the network to allow access to the external service, including your on-premises servers.

This latter point is why you should spend some time thinking about your network configuration. Network addresses and subnets aren't trivial to change once you have them set up. If you plan to connect your private company network to the Azure services, you want to make sure you consider the topology before putting any VMs into place.

When you set up a virtual network, you specify the available address spaces, subnets, and security. If the VNet is connected to other VNets, you must select address ranges that aren't overlapping. This is the range of private addresses that the VMs and services in your network can use. You can use unroutable IP addresses such as 10.0.0.0/8, 172.16.0.0/12, or 192.168.0.0/16, or define your own

range. Azure treats any address range as part of the private VNet IP address space if it's only reachable within the VNet, within interconnected VNets, and from your on-premises location. If someone else is responsible for the internal networks, you should work with that person before selecting your address space to make sure there's no overlap. Let them know what space you want to use, so they don't try to use the same range of IP addresses.

Segregate your network

After deciding the virtual network address space(s), you can create one or more subnets for your virtual network. You create these subnets to break up your network into more manageable sections. For example, you might assign 10.1.0.0 to VMs, 10.2.0.0 to back-end services, and 10.3.0.0 to SQL Server VMs.

Note

Azure reserves the first four addresses and the last address in each subnet for its use.

Secure the network

By default, there's no security boundary between subnets, so services in each of these subnets can talk to one another. However, you can set up Network Security Groups (NSGs), which allow you to control the traffic flow to and from subnets and to and from VMs. NSGs act as software firewalls, applying custom rules to each inbound or outbound request at the network interface and subnet level. This way, you can fully control every network request coming in or out of the VM.

Plan each VM deployment

Once you have mapped out your communication and network requirements, you can start thinking about the VMs you want to create. A good plan is to select a server and take an inventory:

- Which OS is used?
- How much disk space is in use?
- What kind of data does this use? Are there restrictions (legal or otherwise) around how it's stored or where it's physically located?
- What sort of CPU, memory, and disk I/O load does the server have? Is there burst traffic to account for?

We can then start to answer some of the questions Azure has for a new virtual machine.

VM name

The VM name is used as the computer name, which is configured as part of the operating system. You can specify a name of up to 64 characters on a Linux VM and 15 characters on a Windows VM.

This name also defines a manageable **Azure resource**, and it's not trivial to change later. That means you should choose names that are meaningful and consistent, so you can easily identify what the VM does. A good convention is to include the following information in the name:

Element	Example	Notes
Environment	dev, prod, QA	Identifies the environment for the resource
Location	eus for East US, jp for Japan West	Identifies the region into which the resource is deployed
Instance	01, 02	For resources that have more than one named instance (web servers, etc.)
Product or Service	service	Identifies the product, application, or service that the resource supports
Role	sql, web, messaging	Identifies the role of the associated resource

For example, `deveus-webvm01` might represent the first development web server hosted in the East US location.

Decide the location for the VM

Azure has datacenters all over the world filled with servers and disks. These datacenters are grouped into geographic *regions* ('West US', 'North Europe', 'Southeast Asia', etc.) to provide redundancy and availability.

When you create and deploy a virtual machine, you must select a region where you want to allocate the resources. You can place your VMs as close as possible to your users to improve performance and to meet any legal, compliance, or tax requirements.

Two other things to think about regarding the location choice. First, the location can limit your available options. Each region has different hardware available and some configurations aren't available in all regions. Second, there are price differences between locations. If your workload isn't bound to a specific location, it can be very cost effective to check your required configuration in multiple regions to find the lowest price.

Determine the size of the VM

Once you have the name and location set, you need to decide on the [size of your VM](#). Rather than specify processing power, memory, and storage capacity independently, Azure provides different *VM sizes* that offer variations of these elements in different sizes. Azure provides a wide range of VM size options allowing you to select the appropriate mix of compute, memory, and storage for what you want to do.

The best way to determine the appropriate VM size is to consider the type of workload your VM needs to run. Based on the workload, you're able to choose from a subset of available VM sizes. Workload options are classified as follows on Azure:

Option	Description
General purpose	General-purpose VMs are designed to have a balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers.
Compute optimized	Compute optimized VMs are designed to have a high CPU-to-memory ratio. Suitable for medium traffic web servers, network appliances, batch processes, and application servers.
Memory optimized	Memory optimized VMs are designed to have a high memory-to-CPU ratio. Great for relational database servers, medium to large caches, and in-memory analytics.
Storage optimized	Storage optimized VMs are designed to have high disk throughput and IO. Ideal for VMs running databases.
GPU	GPU VMs are specialized virtual machines targeted for heavy graphics rendering and video editing. These VMs are ideal options for model training and inferencing with deep learning.
High performance compute	High performance compute is the fastest and most powerful CPU virtual machines with optional high-throughput network interfaces.

You're able to filter on the workload type when you configure the VM size in the Azure. The size you choose directly affects the cost of your service. The more CPU, memory, and GPU you need, the higher the price point.

What if my size needs change?

Azure allows you to change the VM size when the existing size no longer meets your needs. You can upgrade or downgrade the VM, as long as your current hardware configuration is allowed in the new size. The ability to change VM size provides a fully agile and scalable approach to VM management.

The VM size can be changed while the VM is running, as long as the new size is available in the current hardware cluster the VM is running on. The Azure portal makes the size options obvious by only showing you available size choices. The command line tools report an error if you attempt to resize a VM to an unavailable size. Changing a running VM size automatically reboots the machine to complete the request.

If you stop and deallocate the VM, you can then select any size available in your region since deallocation removes your VM from the cluster it was running on.

Warning

Be careful about resizing production VMs - they will be rebooted automatically which can cause a temporary outage and change some configuration settings such as the IP address.

Parts of a VM and how they're billed

When you create a virtual machine, you're also creating resources that support the virtual machine. These resources come with their own costs that should be considered.

The default resources supporting a virtual machine and how they're billed are detailed in the following table:

Resource	Description	Cost
Virtual network	For giving your virtual machine the ability to communicate with other resources	Virtual Network pricing
A virtual Network Interface Card (NIC)	For connecting to the virtual network	There is no separate cost for NICs. However, there is a limit to how many NICs you can use based on your VM's size . Size your VM accordingly and reference Virtual Machine pricing .
A private IP address and sometimes a public IP address.	For communication and data exchange on your network and with external networks	IP Addresses pricing
Network security group (NSG)	For managing the network traffic too and from your VM. For example, you might need to open port 22 for SSH access, but you might want to block traffic to port 80. Blocking and allowing port access is done through the NSG.	There are no additional charges for network security groups in Azure.
OS Disk and possibly separate disks for data.	It's a best practice to keep your data on a separate disk from your operating system, in case you ever have a VM fail, you can simply detach the data disk, and attach it to a new VM.	All new virtual machines have an operating system disk and a local disk. Azure doesn't charge for local disk storage. The operating system disk, which is usually 127GiB but is smaller for some images, is charged at the regular rate for disks . You can see the cost for attach Premium (SSD based) and Standard (HDD) based disks to your virtual machines on the Managed Disks pricing page .
In some cases, a license for the OS	For providing your virtual machine runs to run the OS	The cost varies based on the number of cores on your VM, so size your VM accordingly . The cost can be reduced through the Azure Hybrid Benefit .

Understanding the pricing model

There are two separate costs the subscription is charged for every VM: compute and storage. By separating these costs, you scale them independently and only pay for what you need.

Compute costs - Compute expenses are priced on a per-hour basis but billed on a per-minute basis. For example, you're only charged for 55 minutes of usage if the VM is deployed for 55 minutes. You're not charged for compute capacity if you stop and deallocate the VM since deallocation releases the hardware. The hourly price varies based on the VM size and OS you select. Linux-based instances are cheaper because there's no operating system license charge. For Windows, the cost for a VM includes the charge for the operating system.

Tip

You might be able to save money by reusing existing licenses with the **Azure Hybrid benefit** for [Linux](#) or [Windows](#).

You're able to choose from two payment options for compute costs.

Option	Description
Pay as you go	With the pay-as-you-go option, you pay for compute capacity by the second, with no long-term commitment or upfront payments. You're able to increase or decrease compute capacity on demand and start or stop at any time. Select this option if you run applications with short-term or unpredictable workloads that can't be interrupted. For example, if you're doing a quick test, or developing an app in a VM, pay-as-you-go is the appropriate option.
Reserved Virtual Machine Instances	The Reserved Virtual Machine Instances (RI) option is an advance purchase of a virtual machine for one or three years in a specified region. The commitment is made up front, and in return, you get up to 72% price savings compared to pay-as-you-go pricing. RI s are flexible and can easily be exchanged or returned for an early termination fee. Select this option if the VM has to run continuously, or you need budget predictability, and you can commit to using the VM for at least a year.

Storage costs - You're charged separately for the storage the VM uses. The status of the VM has no relation to the storage charges that are incurred. If the VM is stopped/deallocated and you aren't billed for the running VM, you're still charged for the storage used by the disks.

Storage for the VM

All Azure virtual machines have at least two virtual hard disks (VHDs). The first disk stores the operating system, and the second is used as temporary storage. You should add more data disks to store application data. Separating out the data to different disks allows you to manage the disks independently. The VM size determines the maximum number of data disks you can attach to your VM, typically two per vCPU.

There are five disk types, each intended to address a specific customer scenario:

- [Ultra disks](#)

- [Premium SSD v2 \(preview\)](#)
- [Premium SSDs \(solid-state drives\)](#)
- [Standard SSDs](#)
- [Standard HDDs \(hard disk drives\)](#)

The following table provides a comparison of the five disk types to help you decide which to use.

	Ultra disk	Premium SSD v2	Premium SSD	Standard SSD	Standard HDD
Disk type	SSD	SSD	SSD	SSD	HDD
Scenario	IO-intensive workloads such as SAP HANA , top tier databases (for example, SQL, Oracle), and other transaction-heavy workloads.	Production and performance-sensitive workloads that consistently require low latency and high IOPS and throughput	Production and performance sensitive workloads	Web servers, lightly used enterprise applications and dev/test	Backup, noncritical, infrequent access
Max disk size	65,536 gibibytes (GiB)	65,536 GiB	32,767 GiB	32,767 GiB	32,767 GiB
Max throughput	4,000 MB/s	1,200 MB/s	900 MB/s	750 MB/s	500 MB/s
Max IOPS	160,000	80,000	20,000	6,000	2,000
Usable as OS Disk?	No	No	Yes	Yes	Yes

Select an operating system

Azure provides various OS images that you can install into the VM, including many Linux distributions. The choice of OS might influence your hourly compute pricing as Azure bundles the cost of the OS license into the price.

If you're looking for more than just base OS images, you can search the Azure Marketplace for more sophisticated images that include the OS and popular software tools for specific scenarios. For example, if you needed a new WordPress site, the standard technology stack would consist of a Linux server, Apache web server, a MySQL database, and PHP. Instead of setting up and configuring each component, you can use a Marketplace image and install the entire stack all at once.

Finally, if you can't find a suitable OS image, you can create your own image with what you need, and use them to create VMs. You can create individual images for use in development and test. Or, you can create an [Azure Compute Gallery](#) to manage multiple images and replicate them to the regions where they're needed.

3. Exercise - Create a VM using the Azure portal

<https://learn.microsoft.com/en-us/training/modules/intro-to-azure-virtual-machines/3-create-a-vm>

Exercise - Create a VM using the Azure portal

Completed

You've planned out the network infrastructure and identified a few VMs to migrate to the cloud. You have several choices for creating your VMs. The choice you make depends on the environment you're comfortable with. Azure supports a web-based portal for creating and administering resources. You can also choose to use command-line tools that run on Linux, macOS, and Windows.

Note

This exercise is optional. If you want to complete this exercise, you'll need to create an Azure subscription before you begin. If you don't have an Azure account or you don't want to create one at this time, you can read through the instructions so you understand the information that's being presented.

Note

You need to use a resource group to complete the steps in this exercise. You can use a resource group that you already created, or you can create a new resource group specifically for this exercise. If you choose to create a new resource group, that will make it easier to clean up any resources that you create as you complete the exercise. If you don't have an existing resource group or you want to create a new one specifically for this exercise, you can follow the steps in [Use the Azure portal and Azure Resource Manager to manage resource groups](#) to create a resource group by using the Azure portal, or you can follow the steps in [Manage Azure resource groups by using Azure CLI](#) to create a resource group by using the the Azure CLI.

Note

Throughout this exercise, replace **myResourceGroupName** in the examples with the name of an existing resource group, or the name of the resource group that you created for this exercise.

Options to create and manage VMs

Let's explore the Azure portal first - it's the easiest way to start with Azure.

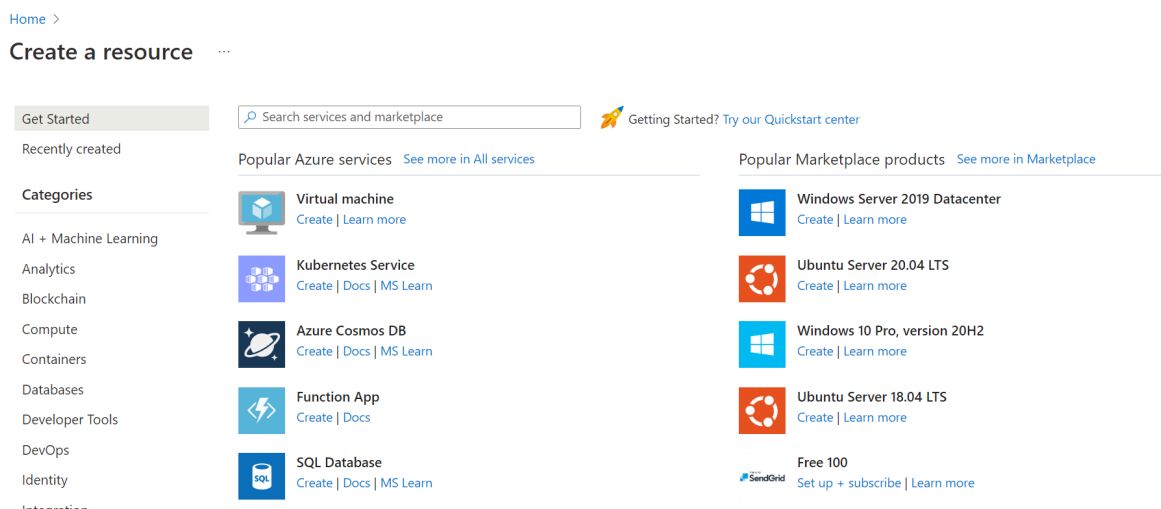
Azure portal

The **Azure portal** provides an easy-to-use browser-based user interface that enables you to create and manage all your Azure resources. For example, you can set up a new database, increase the compute power of your virtual machines, and monitor your monthly costs. It's also a great learning tool, because you can survey all available resources and use guided wizards to create the ones you need.

Create an Azure VM with the Azure portal

Let's assume you want to create a VM running a web server on Ubuntu. Setting up a site isn't difficult, but there are a couple of things to keep in mind. You need to install and configure an operating system, configure a website, install a database, and worry about things like firewalls. We're going to cover creating VMs in the next few modules, but let's create one here to see how easy it is. We don't go through all the options - check out one of the **Create a VM** modules to get complete details on each option.

1. Sign in to the [Azure portal](#).
2. On the Azure home page, under **Azure services**, select **Create a resource**. The **Create a resource** pane appears, displaying popular products for Azure services.



3. We want to create a VM, so select **Virtual machine**.

4. The **Create virtual machine** pane appears.

Configure the VM

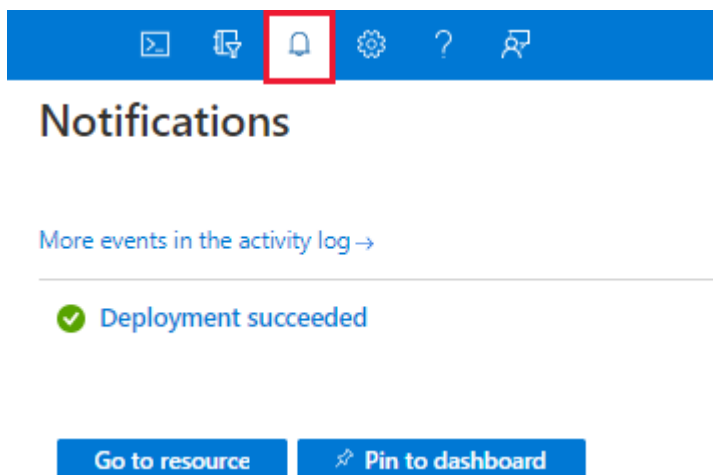
You need to configure the basic parameters of your virtual machine. If some of the options at this point are unfamiliar to you, that's OK. We're going to describe all of these options in a future module. You're welcome to copy the values used here.

1. On the **Basics** tab, enter the following values for each setting.

Setting	Value
Project details	
Subscription	Select your subscription
Resource group	Select myResourceGroupName from the drop-down
Instance details	
Virtual machine name	Enter <i>test-ubuntu-cus-vm</i>
Region	From the dropdown list, select a geographical location close to you.
Availability options	No infrastructure redundancy required
Security type	Standard
Image	Ubuntu Server 24.04 LTS - Gen2
VM architecture	x64
Run with Azure Spot discount	Unchecked
Size	Standard D2s V3
Administrator account	
Authentication type	SSH public key
Username	Enter a username
SSH public key source	Generate a new key pair
Key pair name	test-ubuntu-cus-vm_key
Inbound port rules	
Public inbound ports	Allow selected ports
Select inbound ports	SSH (22)

2. There are several other tabs you can explore to see the settings you can influence during the VM creation. After you're finished exploring, select **Review + create** to review and validate the settings.

- Azure validates your configuration settings for a resource before it creates it. You might need to supply some additional information based on the requirements of the image creator built into Azure. It's simple; just open the tab that has an error. Verify all the settings are set the way you want, and then select **Create** to deploy and create the VM.
- The **Generate new key pair** window opens. Select **Download private key and create resource**.
- You can monitor the deployment in the **Deployment details** on the **Overview** pane or through the **Notifications** pane. Select the notifications icon in the top right toolbar to show or hide the Notifications pane.



The VM deployment process takes a few minutes to complete. You receive a notification informing you that the deployment succeeded.

- Select **Go to resource**. The **Overview** page of your VM appears.

Here, you can see all the information and configuration options for your newly created Ubuntu VM. One of the pieces of information is the **Public IP address**.

The image shows the Azure portal's Overview page for a Virtual Machine. The page is divided into two main sections. The left section, titled "Essentials", contains metadata about the resource group and subscription. The right section, titled "Properties", contains details about the VM's configuration. The "Networking" section is highlighted with a red box, showing the "Public IP address" field. The "Size" section is also visible, showing the VM's size and vCPUs. The "Disk" section is partially visible at the bottom.

Essentials	
Resource group (change)	MyResourceGroup
Status	Running
Location	West US
Subscription (change)	Concierge Subscription
Subscription ID	aaaa0a0a-bb1b-cc2c-dd3d-eeeeee4e4e4e
Tags (change)	Click here to add tags

Properties	
Virtual machine	
Computer name	test-ubuntu-cus-vm
Operating system	Linux (ubuntu 18.04)
Publisher	Canonical
Offer	UbuntuServer
Plan	18.04-LTS
VM generation	V1
Agent status	Ready
Agent version	2.3.1.1
Host group	None
Host	-
Proximity placement group	-
Colocation status	N/A

Networking	
Public IP address	xx.xxx.xx.xx
Public IP address (IPv6)	-
Private IP address	10.0.0.4
Private IP address (IPv6)	-
Virtual network/subnet	VNet1/default
DNS name	Configure

Size	
Size	Standard D2s v3
vCPUs	2
RAM	8 GiB

Disk	
------	--

When you enabled SSH public key authentication in an earlier step, the user interface also gave an option to enable SSH. SSH allows you to connect to your VM via the public IP using any SSH client.

Congratulations! With a few steps, you deployed a VM that runs Linux. Let's explore some other ways we could have created a VM.

4. Describe the options available to create and manage an Azure Virtual Machine

<https://learn.microsoft.com/en-us/training/modules/intro-to-azure-virtual-machines/4-describe-other-create-vm-options>

Describe the options available to create and manage an Azure Virtual Machine

Completed

The Azure portal is the easiest way to create resources such as VMs when you're getting started. However, it's not necessarily the most efficient or quickest way to work with Azure, particularly if you need to create several resources together. In our case, we eventually create dozens of VMs to handle different tasks. Creating them manually in the Azure portal wouldn't be a fun task!

Let's look at some other ways to create and administer resources in Azure:

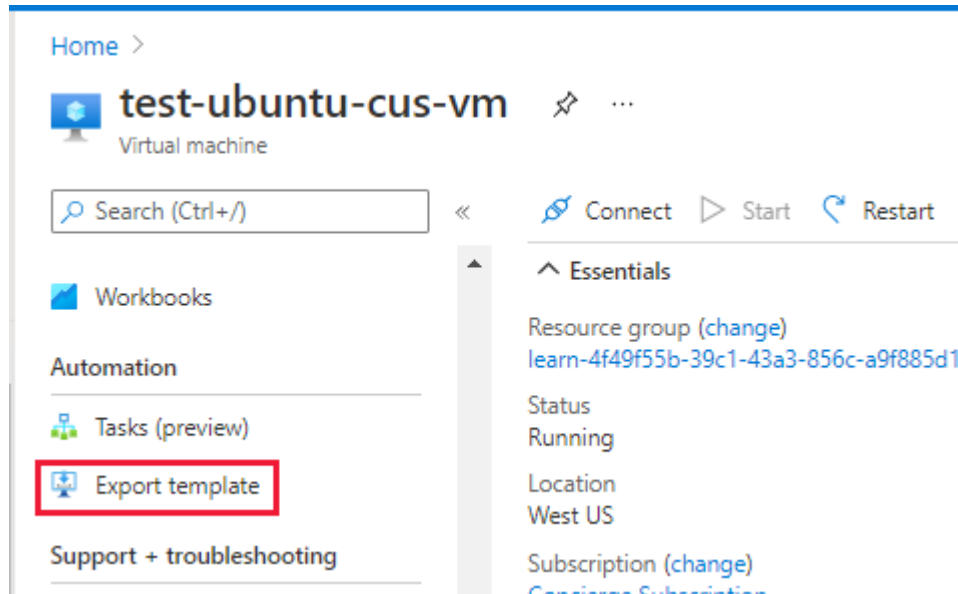
- Azure Resource Manager templates
- Azure PowerShell
- Azure CLI
- Azure REST API
- Azure Client SDK
- Azure VM Extensions
- Azure Automation Services

Resource Manager templates

Let's assume you want to create a copy of a VM with the same settings. You could create a VM image, upload it to Azure, and reference it as the basis for your new VM. This process is inefficient and time-consuming. Azure provides you with the option to create a template from which to create an exact copy of a VM.

Resource Manager templates are JSON files that define the resources you need to deploy for your solution.

You can create a resource template for your VM. From the VM menu, under **Automation** select **Export template**.



Note

The policies for the resources included in the sandbox for this Learn module prevent you from being able to export the VM you just created; that said, an exported template is an easy-to-edit JSON file. You have the option to download or save a template for later use, or immediately deploy a new VM based on the template. For example, you might create a VM from a template in a test environment, and find it doesn't quite work to replace your on-premises machine. You can delete the resource group, which deletes all of the resources, tweak the template, and try again. If you only want to make changes to the existing deployed resources, you can change the template used to create it, and redeploy it. Resource Manager will change the resources to match the new template.

After you have it working the way you want it, you can use that template to easily replicate multiple versions of your infrastructure, such as staging and production. You can parameterize fields such as the VM name, network name, storage account name, and so on, and load the template repeatedly, using different parameters to customize each environment.

For more information about using templates, see [Quickstart: Create an Ubuntu Linux virtual machine using an ARM template](#).

Azure CLI

An option for scripting and command-line Azure interaction is the **Azure CLI**.

The Azure CLI is Microsoft's cross-platform command-line tool for managing Azure resources such as virtual machines and disks from the command line. It's available for Linux, macOS, Windows, or in a browser using the Cloud Shell.

For example, from the CLI, you can create an Azure VM with the `az vm create` command.

```
az vm create \  
  --resource-group TestResourceGroup \  
  --name test-wp1-eus-vm \  
  --image Ubuntu2204 \  
  --admin-username azureuser \  
  --generate-ssh-keys
```

The Azure CLI can be used with other scripting languages, like Ruby and Python.

Learn more about creating and managing VMs in the **Manage virtual machines with the Azure CLI tool** module.

For more information about using the Azure CLI to create VMs, see [Quickstart: Create a Linux virtual machine using the CLI](#).

Azure PowerShell

Azure PowerShell is ideal for one-off interactive tasks and/or the automation of repeated tasks.

Note

PowerShell is a cross-platform shell that provides services like the shell window and command parsing. Azure PowerShell is an optional add-on package that adds the Azure-specific commands (referred to as **cmdlets**). You can learn more about installing and using Azure PowerShell in a separate training module.

For example, you can use the `New-AzVM` cmdlet to create a new Debian-based Azure virtual machine.

```
New-AzVm \  
  -ResourceGroupName "TestResourceGroup" \  
  -Name "test-wp1-eus-vm" \  
  -Location "East US" \  
  -Image Debian11 \  
  -VirtualNetworkName "test-wp1-eus-network" \  
  -SubnetName "default" \  
  -SecurityGroupName "test-wp1-eus-nsg" \  
  -PublicIpAddressName "test-wp1-eus-pubip" \  
  -GenerateSshKey \  
  -SshKeyName myPSKey \  
  -OpenPorts 22
```


As shown here, you supply various parameters to handle the large number of VM configuration settings available. Most of the parameters have reasonable values; you only need to specify the required parameters. Learn more about creating and managing VMs with Azure PowerShell in the **Automate Azure tasks using scripts with PowerShell** module.

For more information about using PowerShell to create VMs, see [Quickstart: Create a Linux virtual machine using PowerShell](#).

Terraform

Azure also has a Terraform provider, so you can easily use Terraform to create and manage your VMs. Terraform enables the definition, preview, and deployment of cloud infrastructure. Using Terraform, you create configuration files using HCL syntax. The HCL syntax allows you to specify the cloud provider - such as Azure - and the elements that make up your cloud infrastructure. After you create your configuration files, you create an execution plan that allows you to preview your infrastructure changes before they're deployed. Once you verify the changes, you apply the execution plan to deploy the infrastructure.

For more information, see the [Azure Terraform Provider](#) and [Quickstart: Use Terraform to create a VM](#).

Programmatic (APIs)

Generally speaking, both Azure PowerShell and Azure CLI are good options if you have simple scripts to run and want to stick to command-line tools. When it comes to more complex scenarios, where the creation and management of VMs form part of a larger application with complex logic, another approach is needed.

You can interact with every type of resource in Azure programmatically.

Azure REST API

The Azure REST API provides developers with operations categorized by resource and the ability to create and manage VMs. Operations are exposed as URIs with corresponding HTTP methods (`GET` , `PUT` , `POST` , `DELETE` , and `PATCH`) and a corresponding response.

The Azure Compute APIs give you programmatic access to virtual machines and their supporting resources.

For more information, see the [Virtual Machines REST API reference](#).

Azure Client SDK

Even though the REST API is platform and language agnostic, most often developers look toward a higher level of abstraction. The Azure Client SDK encapsulates the Azure REST API, making it much easier for developers to interact with Azure.

The Azure Client SDKs are available for various languages and frameworks, including .NET-based languages such as C#, Java, Node.js, PHP, Python, Ruby, and Go.

Here's an example snippet of C# code to create an Azure VM using the `Microsoft.Azure.Management.Fluent` NuGet package.

```
var azure = Azure
    .Configure()
    .WithLogLevel(HttpLoggingDelegatingHandler.Level.Basic)
    .Authenticate(credentials)
    .WithDefaultSubscription();
// ...
var vmName = "test-wp1-eus-vm";

azure.VirtualMachines.Define(vmName)
    .WithRegion(Region.USEast)
    .WithExistingResourceGroup("TestResourceGroup")
    .WithExistingPrimaryNetworkInterface(networkInterface)
    .WithLatestWindowsImage("MicrosoftWindowsServer", "WindowsServer", "2012-R2-Datacenter")
    .WithAdminUsername("jenc")
    .WithAdminPassword("aReallyGoodPasswordHere")
    .WithComputerName(vmName)
    .WithSize(VirtualMachineSizeTypes.StandardDS1)
    .Create();
```

Here's the same snippet in Java using the **Azure Java SDK**.

```
String vmName = "test-wp1-eus-vm";
// ...
VirtualMachine virtualMachine = azure.virtualMachines()
    .define(vmName)
    .withRegion(Region.US_EAST)
    .withExistingResourceGroup("TestResourceGroup")
    .withExistingPrimaryNetworkInterface(networkInterface)
    .withLatestWindowsImage("MicrosoftWindowsServer", "WindowsServer", "2012-R2-Datacenter")
    .withAdminUsername("jenc")
    .withAdminPassword("aReallyGoodPasswordHere")
    .withComputerName(vmName)
    .withSize("Standard_DS1")
    .create();
```

Azure VM extensions

Let's assume you want to configure and install more software on your virtual machine after the initial deployment. You want this task to use a specific configuration, monitored and executed automatically.

Azure VM extensions are small applications that enable you to configure and automate tasks on Azure VMs after initial deployment.

For more information, see [Azure virtual machine extensions and features](#).

Azure Automation services

Saving time, reducing errors, and increasing efficiency are some of the most significant operational management challenges faced when managing remote infrastructure. If you have numerous infrastructure services, you might want to consider using higher-level services in Azure to help you operate from a higher level.

Azure Automation enables you to integrate services that allow you to automate frequent, time-consuming, and error-prone management tasks with ease. These services include **process automation**, **configuration management**, and **update management**.

- **Process Automation.** Let's assume you have a VM that is monitored for a specific error event. You want to take action, and fix the problem as soon as it's reported. Process automation enables you to set up watcher tasks that can respond to events that may occur in your datacenter.
- **Configuration Management.** Perhaps you want to track software updates that become available for the operating system that runs on your VM. There are specific updates you may want to include or exclude. Configuration management enables you to track these updates, and take action as required. You use **Microsoft Endpoint Configuration Manager** to manage your company's PC, servers, and mobile devices. You can extend this support to your Azure VMs with Configuration Manager.
- **Update Management.** Use this service to manage updates and patches for your VMs. With this service, you're able to assess the status of available updates, schedule installation, and review deployment results to verify updates applied successfully. Update management incorporates services that provide process and configuration management. You enable update management for a VM directly from your **Azure Automation** account. You can also enable update management for a single virtual machine from the virtual machine pane in the portal.

Auto-shutdown

Auto-shutdown is a feature in Azure that allows you to automatically shut down your VMs on a schedule. Use Auto-shutdown to save costs by ensuring that your VMs are not running when they aren't needed. You can set the schedule for auto-shutdown to occur daily or weekly, and you can also specify the time zone for the schedule.

To navigate to the Auto-shutdown feature in a VM in the Azure portal, go to the VM's blade in the portal, click on "Auto-shutdown" under the "Operations" section, and then configure the auto-shutdown settings according to your preferences.

Home > myVM



myVM | Auto-shutdown

Virtual machine

Search



Save



Discard



Feedback

Operations



Bastion



Auto-shutdown



Backup

Enabled



On



Off

Scheduled shutdown

7:00:00 PM

Time zone

(UTC) Coordinated Universal Time

Send notification before auto-shutdown?



No

For more information, see [Auto-shutdown](#).

As you can see, Azure provides various tools to create and administer resources so that you can integrate management operations into a process *that works for you*. Let's examine some of the other Azure services to make sure your infrastructure resources are running smoothly.

5. Manage the availability of your Azure VMs

<https://learn.microsoft.com/en-us/training/modules/intro-to-azure-virtual-machines/5-high-availability>

Manage the availability of your Azure VMs

Completed

Frequently, the success of a services company is directly related to the service level agreements (SLA) the company has with its customers. Your customers expect the services you provide always to be available, and their data kept safe. This security is something that Microsoft takes seriously. Azure provides tools you can use to manage availability, data security, and monitoring, so you know your services are always available for your customers.

Administration of an Azure VM isn't limited to managing the operating system, or software that runs on the VM. It helps to know which services Azure provides that ensure service availability and support automation. These services help you to plan your organization's business continuity and disaster recovery strategy.

Here, we cover an Azure service that helps you improve VM availability, streamlines VM management tasks, and keeps your VM data backed up and safe. Let's start by defining availability.

What is availability?

Availability is the percentage of time a service is available for use.

Let's assume you have a website, and you want your customers to be able to always access information. Your expectation is 100% availability concerning website access.

Why do I need to think about availability when using Azure?

Azure VMs run on physical servers hosted within the Azure datacenter. As with most physical devices, there's a chance that there could be a failure. If the physical server fails, the virtual machines hosted on that server also fail. If a failure happens, Azure moves the VM to a healthy host server automatically. However, this self-healing migration could take several minutes, during which the application(s) hosted on that VM aren't available.

Periodic updates initiated by Azure itself can also affect the VMs. These maintenance events range from software updates to hardware upgrades and are required to improve platform reliability and performance. These events usually are performed without impacting any guest VMs, but sometimes the virtual machines reboot to complete an update or upgrade.

Availability zones

[Availability zones](#) expands the level of control you have to maintain the availability of the applications and data on your VMs. An Availability Zone is a physically separate zone, within an Azure region. There are three Availability Zones per supported Azure region.

Each Availability Zone has a distinct power source, network, and cooling. By designing your solutions to use replicated VMs in zones, you can protect your apps and data from the loss of a data center. If one zone is compromised, then replicated apps and data are instantly available in another zone.

Virtual Machines Scale Sets

[Azure virtual machine scale sets](#) let you create and manage a group of load balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide high availability to your applications, and allow you to centrally manage, configure, and update many VMs. There's no cost for the scale set itself, you only pay for each VM instance that you create.

Virtual machines in a scale set can also be deployed into multiple availability zones, a single availability zone, or regionally. Availability zone deployment options may differ based on the [orchestration mode](#).

Load balancer

Combine the [Azure Load Balancer](#) with an availability zone or availability set to get the most application resiliency. The Azure Load Balancer distributes traffic between multiple virtual machines. For our Standard tier virtual machines, the Azure Load Balancer is included. Not all virtual machine

tiers include the Azure Load Balancer. For more information about load balancing your virtual machines, see **Load Balancing virtual machines** for [Linux](#) or [Windows](#).

Azure Storage redundancy

Azure Storage always stores multiple copies of your data so that it's protected from planned and unplanned events, including transient hardware failures, network or power outages, and massive natural disasters. Redundancy ensures that your storage account meets its availability and durability targets even in the face of failures.

When deciding which redundancy option is best for your scenario, consider the tradeoffs between lower costs and higher availability. The factors that help determine which redundancy option you should choose include:

- How your data is replicated in the primary region
- Whether your data is replicated to a second region that is geographically distant to the primary region, to protect against regional disasters
- Whether your application requires read access to the replicated data in the secondary region if the primary region becomes unavailable for any reason

For more information, see [Azure Storage redundancy](#).

Failover across locations

You can also replicate your infrastructure across sites to handle regional failover. **Azure Site Recovery** replicates workloads from a primary site to a secondary location. If an outage happens at your primary site, you can fail over to a secondary location. This failover enables users to continue to access your applications without interruption. You can then fail back to the primary location after it's up and running again. Azure Site Recovery is about replication of virtual or physical machines; it keeps your workloads available in an outage.

While there are many attractive technical features to Site Recovery, there are at least two significant business advantages:

- Site Recovery enables the use of Azure as a destination for recovery, thus eliminating the cost and complexity of maintaining a secondary physical datacenter.
- Site Recovery makes it incredibly simple to test failovers for recovery drills without impacting production environments. This feature makes it easy to test your planned or unplanned failovers. After all, you don't have a good disaster recovery plan if you've never tried to fail over.

The recovery plans you create with Site Recovery can be as simple or as complex as your scenario requires. They can include custom PowerShell scripts, Azure Automation runbooks, or manual intervention steps. You can use the recovery plans to replicate workloads to Azure, easily enabling

new opportunities for migration, temporary bursts during surge periods, or development and testing of new applications.

Azure Site Recovery works with Azure resources, or Hyper-V, VMware, and physical servers in your on-premises infrastructure. It can be a key part of your organization's business continuity and disaster recovery (BCDR) strategy by orchestrating the replication, failover, and recovery of workloads and applications if the primary location fails.

6. Back up your virtual machines

<https://learn.microsoft.com/en-us/training/modules/intro-to-azure-virtual-machines/6-backup-services>

Back up your virtual machines

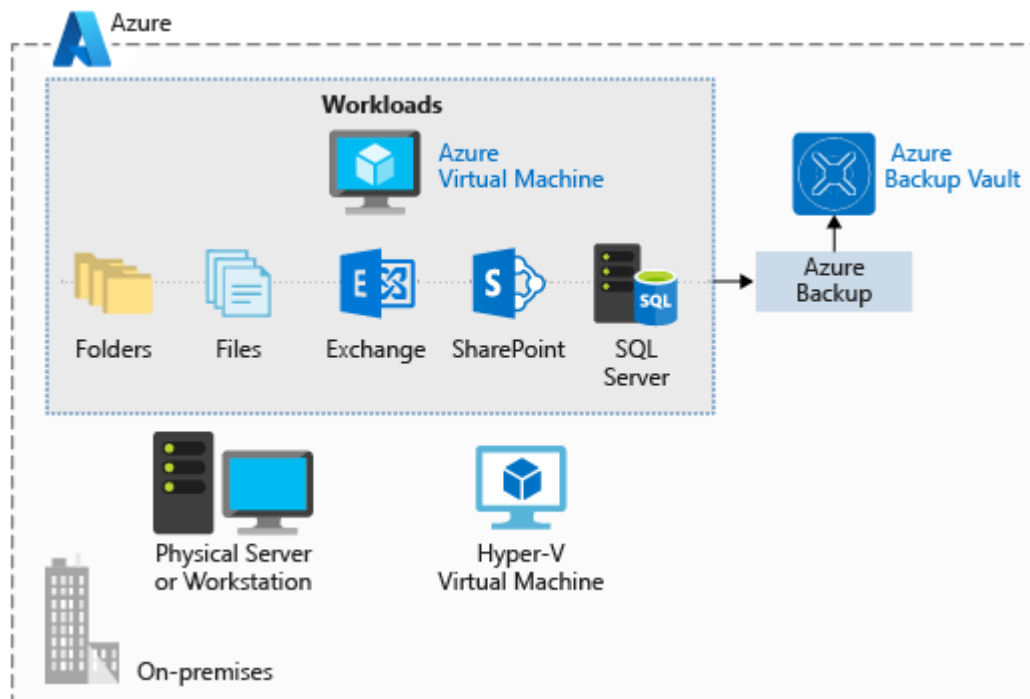
Completed

Data backup and recovery is a necessary piece of the planning for any good infrastructure. Assume a bug erases some company data, or maybe you need to retrieve some archived data for auditing purposes. Maintaining a good backup strategy ensures that you aren't scrambling when data or software needs to be restored.

Azure Backup is a *backup as a service* offering that protects physical or virtual machines no matter where they reside: on-premises or in the cloud.

Azure Backup can be used for a wide range of data backup scenarios, such as:

- Files and folders on Windows OS machines (physical or virtual, local or cloud)
- Application-aware snapshots (Volume Shadow Copy Service)
- Popular Microsoft server workloads such as Microsoft SQL Server, Microsoft SharePoint, and Microsoft Exchange
- Native support for Azure Virtual Machines, both Windows, and Linux
- Linux and Windows 10 client machines



Advantages of using Azure Backup

Traditional backup solutions don't always take full advantage of the underlying Azure platform. The result is a solution that tends to be expensive or inefficient. The solution either offers too much or too little storage, doesn't offer the correct types of storage, or has cumbersome and long-winded administrative tasks. Azure Backup was designed to work in tandem with other Azure services and provides several distinct benefits.

- **Automatic storage management.** Azure Backup automatically allocates and manages backup storage and uses a pay-as-you-use model. You only pay for what you use.
- **Unlimited scaling.** Azure Backup uses the power and scalability of Azure to deliver high availability.
- **Multiple storage options.** Azure Backup offers locally redundant storage where all copies of the data exist within the same region and geo-redundant storage where your data is replicated to a secondary region.
- **Unlimited data transfer.** Azure Backup doesn't limit the amount of inbound or outbound data you transfer. Azure Backup also doesn't charge for the data that is transferred.
- **Data encryption.** Data encryption allows for secure transmission and storage of your data in Azure.
- **Application-consistent backup.** An application-consistent backup means that a recovery point has all required data to restore the backup copy. Azure Backup provides application-consistent backups.
- **Long-term retention.** Azure doesn't limit the length of time you keep the backup data.

Use Azure Backup

Azure Backup uses several components that you download and deploy to each computer you want to back up. The component that you deploy depends on what you want to protect.

- Azure Backup agent
- System Center Data Protection Manager
- Azure Backup Server
- Azure Backup VM extension

Azure Backup uses a Recovery Services vault for storing the backup data. Azure Storage blobs back up a vault, making it an efficient and economical long-term storage medium. With the vault in place, you can select the machines to back up, and define a backup policy (when snapshots are taken and for how long they're stored).

7. Module assessment

<https://learn.microsoft.com/en-us/training/modules/intro-to-azure-virtual-machines/7-knowledge-check>

Module assessment

Completed

8. Summary

<https://learn.microsoft.com/en-us/training/modules/intro-to-azure-virtual-machines/8-summary>

Summary

Completed

In this module, you looked at the decisions you need to make before creating a virtual machine. These decisions include aspects such as the VM size, types of disks used, operating system image selected, and the types of resources created.

You also looked at the options to create and manage virtual machines in Azure. You saw how easy it is to create and manage VMs using the portal. You also learned when to use Resource Manager

templates, PowerShell, the Azure CLI, and the Azure Client SDK.

Finally, you looked at the extensions and services available to more easily administer your VMs.

Important

In the optional exercises for this module, you created resources by using your own Azure subscription. Clean up these resources so that you won't continue to be charged for them.