✓ 100 XP ▶

Introduction

3 minutes

You have been hired by a global auto racing company to modernize their entire monitoring and web platform. They have decided to replace existing Linux servers with a variety of cloud-based infrastructure that leverages the latest in architectural trends. Part of the system will run on the Azure serverless platform using Azure Functions to process real-time race data, pushing statistics, race data, and other relevant bits of analyzed information into clusters of databases. They want to keep their existing website, which was just rewritten last year, but have it connect into this modern data stream.

The website is running on Apache with Linux, and since it's already up and running, you decide to move it directly into Azure by leveraging an Azure virtual machine. This will give the website access to the data with a minimal amount of work on your part.

Learning objectives

In this module, you will:

- Understand the options that are available for virtual machines in Azure
- Create a Linux virtual machine using the Azure portal
- Connect to a running Linux virtual machine using SSH
- Install software and change the network configuration on a VM using the Azure portal

Next unit: Create a Linux virtual Machine in Azure

Continue >

✓ 100 XP ▶

Create a Linux virtual Machine in Azure

10 minutes

We have an existing website running on a local Ubuntu Linux server. Our goal is to create an Azure virtual machine (VM) using the latest Ubuntu image and then migrate the site to the cloud. In this unit, you will learn about the options you will need to evaluate when creating a virtual machine in Azure.

Introduction to Azure Virtual Machines

Azure Virtual Machines is an on-demand, scalable cloud-computing resource. They include processors, memory, storage, and networking resources. You can start and stop virtual machines at will and manage them from the Azure portal or with the Azure CLI. You can also use a remote Secure Shell (SSH) to connect directly to the running VM and execute commands as if you were on a local computer.

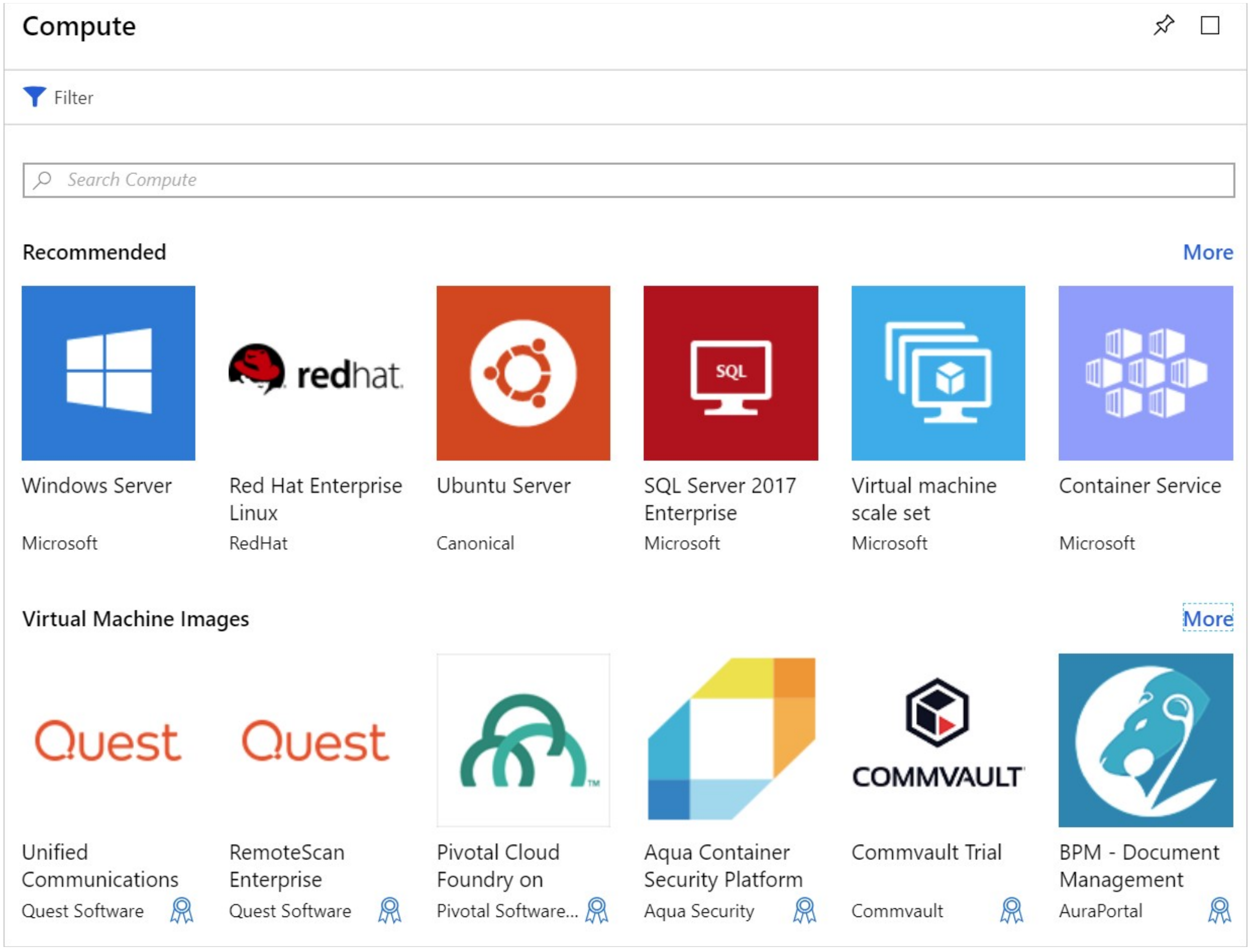
Running Linux in Azure

Creating Linux-based VMs in Azure is easy. Microsoft has partnered with prominent Linux vendors to ensure their distributions are optimized for the Azure platform. You can create virtual machines from prebuilt images for a variety of popular Linux distributions, such as SUSE, Red Hat, and Ubuntu, or build your own Linux distribution to run in the cloud.

Creating an Azure VM

VMs can be defined and deployed on Azure in several ways: the Azure portal, a script (using the Azure CLI or Azure PowerShell), or an Azure Resource Manager template. In all cases, you will need to supply several pieces of information that we'll cover shortly.

The Azure Marketplace also provides preconfigured images that include both an OS and favorite software tools installed for specific scenarios.



Resources used in a Linux VM

When creating a Linux VM in Azure, you also create resources to host the VM. These resources work together to virtualize a computer and run the Linux operating system. These must exist (and be selected during VM creation), or they will be created with the VM:

- A virtual machine that provides CPU and memory resources
- An Azure Storage account to hold the virtual hard disks
- Virtual disks to hold the OS, applications, and data
- A virtual network (VNet) to connect the VM to other Azure services or your on-premises hardware
- A network interface to communicate with the VNet

- An optional public IP address so you can access the VM

Like other Azure services, you'll need a **Resource Group** to contain the VM (and optionally group these resources for administration). When you create a new VM, you can either use an existing resource group or create a new one.

Choose the VM image

Selecting an image is one of the first and most important decisions you'll make when creating a VM. An image is a template that's used to create a VM. These templates include an OS and often other software, such as development tools or web hosting environments.

Anything that a computer can have installed can be included in an image. You can create a VM from an image that's preconfigured to precisely the tasks you need, such as hosting a web app on the Apache HTTP Server.

> **Tip**
>
> You can also create and upload custom disk images.

Sizing your VM

Just as a physical machine has a certain amount of memory and CPU power, so does a virtual machine. Azure offers a range of VMs of differing sizes at different price points. The size that you choose will determine the VM's processing power, memory, and maximum storage capacity.

> ⚠ **Warning**
>
> There are quota limits on each subscription that can impact VM creation. If you run into these quota limits you can open an online customer support request to increase your limits.

VM sizes are grouped into categories, starting with the B-series for basic testing and running up to the H-series for massive computing tasks. You should select the size of the VM based on the workload you want to perform. It is possible to change the size of a VM after it's been created, but the VM must be stopped first. So, it's best to size it appropriately from the start if possible.

Here are some guidelines based on the scenario you are targeting

What are you doing?	Consider these sizes
General use computing/web: Testing and development, small to medium databases, or low to medium traffic web servers.	B, Dsv3, Dv3, DSv2, Dv2
Heavy computational tasks: Medium traffic web servers, network appliances, batch processes, and application servers.	Fsv2, Fs, F
Large memory usage: Relational database servers, medium to large caches, and in-memory analytics.	Esv3, Ev3, M, GS, G, DSv2, Dv2
Data storage and processing: Big data, SQL, and NoSQL databases that need high disk throughput and I/O.	Ls
Heavy graphics rendering or video editing, as well as model training and inferencing (ND) with deep learning.	NV, NC, NCv2, NCv3, ND
High-performance computing (HPC): Your workloads need the fastest and most powerful CPU virtual machines with optional high-throughput network interfaces.	H

Choosing storage options

The next set of decisions revolves around storage. First, you can choose the disk technology. Options include a traditional platter-based hard disk drive (HDD) or a more modern solid-state drive (SSD). Just like the hardware you purchase, SSD storage costs more but provides better performance.

💡 Tip

There are two levels of SSD storage available: standard and premium. Choose Standard SSD disks if you have normal workloads but want better performance. Choose Premium SSD disks if you have I/O intensive workloads or mission-critical systems that need to process data very quickly.

Mapping storage to disks

Azure uses virtual hard disks (VHDs) to represent physical disks for the VM. VHDs replicate the logical format and data of a disk drive but are stored as page blobs in an Azure Storage account. You can choose on a per disk basis what type of storage it should use (SSD or HDD). This allows you to control the performance of each disk, likely based on the I/O you plan to perform on it.

By default, two virtual hard disks (VHDs) will be created for your Linux VM:

1. The **operating system disk**: This is your primary drive, and it has a maximum capacity of 2048 GB. It will be labeled as `/dev/sda` by default.

2. A **temporary disk**: This provides temporary storage for the OS or any apps. On Linux virtual machines, the disk is `/dev/sdb` and is formatted and mounted to `/mnt` by the Azure Linux Agent. It is sized based on the VM size and is used to store the swap file.

⚠ Warning

The temporary disk is not persistent. You should only write data to this disk that is not critical to the system.

What about data?

You can store data on the primary drive along with the OS, but a better approach is to create dedicated *data disks*. You can create and attach additional disks to the VM. Each disk can hold up to 32,767 gibibytes (GiB) of data, with the maximum amount of storage determined by the VM size you select.

> ⓘ **Note**
>
> An interesting capability is to create a VHD image from a real disk. This allows you to easily migrate *existing* information from an on-premises computer to the cloud.

Unmanaged vs. managed disks

The final storage choice you'll make is whether to use **unmanaged** or **managed** disks.

With unmanaged disks, you are responsible for the storage accounts that are used to hold the VHDs that correspond to your VM disks. You pay the storage account rates for the amount of space you use. A single storage account has a fixed rate limit of 20,000 I/O operations/sec. This means that a single storage account is capable of supporting 40 standard virtual hard disks at full throttle. If you need to scale out, then you need more than one storage account, which can get complicated.

Managed disks are the newer and recommended disk storage model. They elegantly solve this complexity by putting the burden of managing the storage accounts onto Azure. You specify the disk type (Premium or Standard) and the size of the disk, and Azure creates and manages both the disk *and* the storage it uses. You don't have to worry about storage account limits, which makes them easier to scale out. They also offer several other benefits:

- **Increased reliability**: Azure ensures that VHDs associated with high-reliability VMs will be placed in different parts of Azure Storage to provide similar levels of resilience.
- **Better security**: Managed disks are real managed resources in the resource group. This means they can use role-based access control to restrict who can work with the VHD data.
- **Snapshot support**: Snapshots can be used to create a read-only copy of a VHD. You have to shut down the owning VM, but creating the snapshot only takes a few seconds. Once it's done, you can power on the VM and use the snapshot to create

a duplicate VM to troubleshoot a production issue or roll back the VM to the point in time that the snapshot was taken.
- **Backup support**: Managed disks can be automatically backed up to different regions for disaster recovery with Azure Backup without affecting the service of the VM.

Network communication

Virtual machines communicate with external resources using a virtual network (VNet). The VNet represents a private network in a single region that your resources communicate on. A virtual network is just like the networks you manage on-premises. You can divide them up with subnets to isolate resources, connect them to other networks (including your on-premises networks), and apply traffic rules to govern inbound and outbound connections.

Planning your network

When you create a new VM, you will have the option of creating a new virtual network or using an existing VNet in your region.

Having Azure create the network together with the VM is simple, but it's likely not ideal for most scenarios. It's better to plan your network requirements *up front* for all the components in your architecture and create the VNet structure separately. Then, create the VMs and place them into the already-created VNets. We'll look more at virtual networks later in this module.

Before we create a virtual machine, we need to decide how we'd like to administer the VM. Let's look at our options.

Next unit: Exercise - Decide an authentication method for SSH

Continue >

✓ 100 XP

Exercise - Create a Linux virtual machine with the Azure portal

20 minutes

This module requires a sandbox to complete. You have used 2 of 10 sandboxes for today. More sandboxes will be available tomorrow.

Activate sandbox

Recall that our goal is to move an existing Linux server running Apache to Azure. We'll start by creating an Ubuntu Linux server.

Create a new Linux virtual machine

We can create Linux VMs with the Azure portal, the Azure CLI, or Azure PowerShell. The easiest approach when you are starting with Azure is to use the portal because it walks you through the required information and provides hints and helpful messages during the creation:

1. Sign into the Azure portal using the same account you activated the sandbox with.

2. On the Azure portal menu or from the **Home** page, select **Create a resource**.

3. In the search box, enter **Ubuntu Server**.

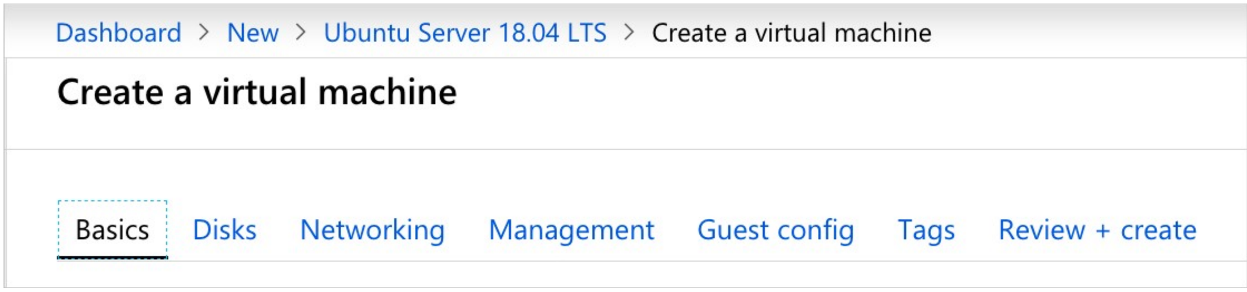
4. Under the **Marketplace** result, select the **All results** link on the top right-hand side to see the different versions available.

5. Select **Ubuntu Server 18.04 LTS Canonical** from the presented list.

6. Click the **Create** button to start configuring the VM.

Configure the VM settings

The VM creation experience in the portal is presented in a wizard format to walk you through all the configuration areas for the VM. Clicking the **Next** button will take you to the next configurable section. However, you can move between the sections at will with the tabs running across the top that identify each part.



Once you fill in all the required options (identified with red asterisks), you can skip the remainder of the wizard experience and start creating the VM through the **Review + Create** button at the bottom.

We'll start with the **Basics** section. These instructions are for the Sandbox portal. If you are using another Azure portal account, you may need to adapt some details accordingly.

Configure basic VM settings

1. For **Subscription**, the sandbox subscription should be selected for you by default.

2. For **Resource group**, the resource group with the name **[sandbox resource group name]** should be selected for you by default.

3. In the **Instance details** section, enter a name for your web server VM, such as **test-web-eus-vm1**. This indicates the environment (**test**), the role (**web**), location (**East US**), service (**vm**), and instance number (**1**).

    - It's considered best practice to standardize your resource names, so you can quickly identify their purpose. Linux VM names must be between 1 and 64 characters and be comprised of numbers, letters, and dashes.

    ⓘ **Note**

> As you change settings and tab out of each free-text field, Azure will validate each value automatically and place a green check mark next to it when it's good. You can hover your mouse pointer over error indicators to get more information on issues it discovers.

4. Select a location.

   The free sandbox allows you to create resources in a subset of the Azure global regions. Select a region from the following list when you create resources:

   - West US 2
   - South Central US
   - Central US
   - East US
   - West Europe
   - Southeast Asia
   - Japan East
   - Brazil South
   - Australia Southeast
   - Central India

5. Set **Availability options** to **No infrastructure redundancy required**. This option can be used to ensure the VM is highly available by grouping multiple VMs together as a set to deal with planned or unplanned maintenance events or outages. For this exercise we will not need this service.

6. Ensure that the image is set to **Ubuntu Server 18.04 LTS**. You can open the drop-down list to see all the options available.

7. Leave the **Size** field with the default of **D2s v3** choice, which gives you two vCPUs with 8 GB of RAM.

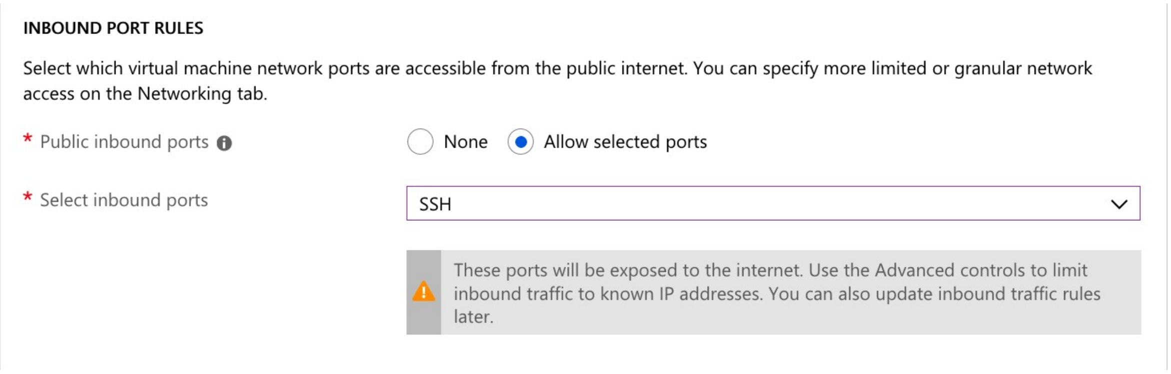
8. Moving on to the **Administrator account** section, for **Authentication type** select the **SSH public key** option.

9. Enter a **username** you'll use to sign in with SSH. Choose something you can remember or write it down.

10. Copy the SSH key from your public key file you created in the previous unit and paste it into the **SSH public key** field.

> ⓘ Important
>
> When you copy the public key into the Azure portal, make sure not to add any additional whitespace or line-feed characters.

11. In the **INBOUND PORT RULES** section, first select **Allow selected ports**. Since this is a Linux VM, we want to be able to access the VM using SSH remotely. Scroll the **Select inbound ports** list if necessary until you find **SSH (22)** and enable it.



INBOUND PORT RULES

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

* Public inbound ports ⓘ ○ None ● Allow selected ports

* Select inbound ports SSH

⚠ These ports will be exposed to the internet. Use the Advanced controls to limit inbound traffic to known IP addresses. You can also update inbound traffic rules later.

Configure disks for the VM

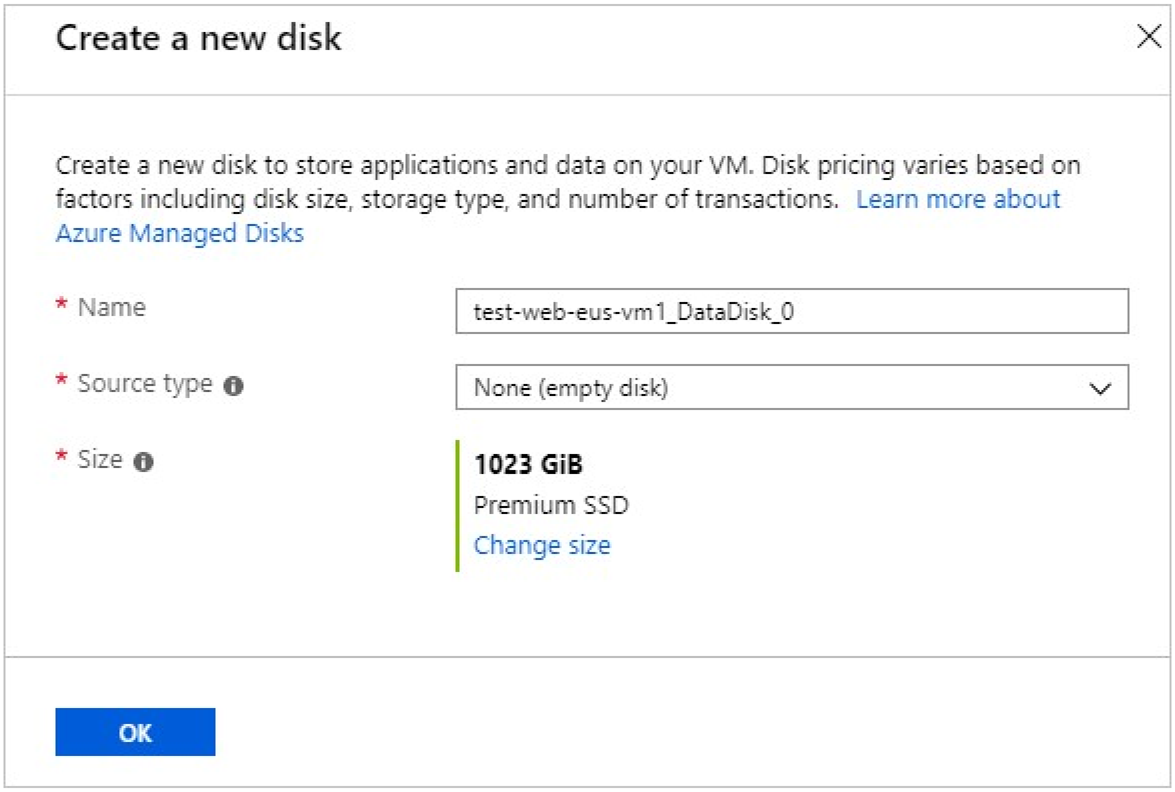
1. Click **Next: Disks >** to move to the **Disks** section.

2. Choose **Premium SSD** for the **OS disk type**.

Create a data disk

Recall that we will get an OS disk (/dev/sda) and a temporary disk (/dev/sdb). Let's add a data disk as well:

1. Click the **Create and attach a new disk** link in the **Data disks** section.



Create a new disk

Create a new disk to store applications and data on your VM. Disk pricing varies based on factors including disk size, storage type, and number of transactions. Learn more about Azure Managed Disks

* Name: test-web-eus-vm1_DataDisk_0

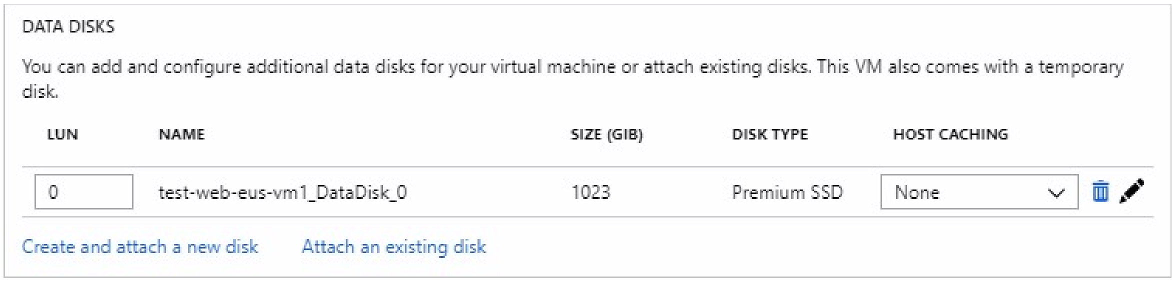
* Source type: None (empty disk)

* Size: **1023 GiB**
Premium SSD
Change size

2. You can take all the defaults: **Premium SSD**, the auto-generated name, size of **1023** GiB, and **None (empty disk)** for **Source type**, although notice that source type is where you could use a snapshot or Azure Blob storage to create a VHD.

3. Click **OK** to create the disk and go back to the **Data disks** section.

4. There should now be a new disk in the first row.



DATA DISKS

You can add and configure additional data disks for your virtual machine or attach existing disks. This VM also comes with a temporary disk.

LUN	NAME	SIZE (GIB)	DISK TYPE	HOST CACHING
0	test-web-eus-vm1_DataDisk_0	1023	Premium SSD	None

Configure the network

1. Click **Next: Networking >** to move to the **Networking** section.

2. In a production environment where we already have other components, you'd want to utilize an *existing* virtual network. That way, your VM can communicate

with the other cloud services in your solution. If there isn't one defined in this location yet, you can create it here and configure the:

- **Address space**: The overall IPV4 space available to this network.
- **Subnets**: The first subnet to subdivide the address space - it must fit within the defined address space. Once the VNet is created, you can add additional subnets.

> **Note**
>
> By default, Azure will create a virtual network, network interface, and public IP for your VM. It's not trivial to change the networking options after the VM has been created, so always double-check the network assignments on services you create in Azure. For this exercise, the defaults should work fine.

Finish configuring the VM and create the image

The rest of the options have reasonable defaults, and there's no need to change any of them. You can explore the other tabs if you like. The individual options have an `(i)` icon next to them that will show a help tip to explain the option. This is a great way to learn about the various options you can use to configure the VM:

1. Click the **Review + create** button at the bottom of the panel.

2. The system will validate your options and give you details about the VM being created.

3. Click **Create** to create and deploy the VM. The Azure dashboard will show the VM that's being deployed. This may take several minutes.

While that's deploying, let's look at what we can do with this VM.

Next unit: Azure virtual machines IP addresses and SSH options

Continue >

✓ 100 XP ▶

Azure virtual machines IP addresses and SSH options

5 minutes

You have created a Linux VM in Azure. The next thing you'll do is configure it for the tasks we want to move to Azure.

Unless you've set up a site-to-site VPN to Azure, your Azure VMs won't be accessible from your local network. If you're just getting started with Azure, it's unlikely that you have a working site-to-site VPN. So how can you connect to the virtual machine?

Azure VM IP addresses

As we saw a moment ago, Azure VMs communicate on a virtual network. They can also have an optional public IP address assigned to them. With a public IP, we can interact with the VM over the Internet. Alternatively, we can set up a virtual private network (VPN) that connects our on-premises network to Azure - letting us securely connect to the VM without exposing a public IP. This approach is covered in another module and is fully documented if you are interested in exploring that option.

Public IP addresses in Azure are dynamically allocated by default. That means the IP address can change over time - for VMs the IP address assignment happens when the VM is restarted. You can pay more to assign static addresses, if you want to connect directly to an IP address and need to ensure that the IP address will not change.

Acknowledging these restrictions, and the alternatives described above, we will use the public IP address of the VM in this module.

Connecting to the VM with SSH

To connect to the VM via SSH, you need:

- the public IP address of the VM
- the username of the local account on the VM
- a public key configured in that account
- access to the corresponding private key
- port 22 open on the VM

Previously, you generated an SSH key pair, and added the public key to the VM configuration, and ensured that port 22 was open.

In the next unit, you'll use this information to open a secure terminal on the VM using SSH.

Once the terminal is open, you have access to all of the standard Linux command-line tools.

Next, let's connect to the VM using SSH.

Next unit: Exercise - Connect to a Linux virtual machine with SSH

Continue ›

✓ 100 XP

Network and security settings

10 minutes

Making adjustments to server configuration is commonly performed with equipment in your on-premises environment. In this sense, you can consider Azure VMs to be an extension of that environment. You can alter configuration, manage networks, open or block traffic, and more through the Azure portal, the Azure CLI, or Azure PowerShell tools.

We've got our server running, and Apache is installed and serving up pages. Our security team mandates that we lock down all our servers, and we've not done anything to this VM yet. We didn't do anything, and it let Apache listen on port 80. Let's explore the Azure network configuration to see how to use the built-in security support to harden our server.

Opening ports in Azure VMs

By default, new VMs are locked down.

Apps can make outgoing requests, but the only inbound traffic allowed is from the virtual network (e.g., other resources on the same local network) and from Azure Load Balancer (probe checks).

There are two steps to adjusting the configuration to support different protocols on the network. When you create a new VM, you have an opportunity to open a few common ports (RDP, HTTP, HTTPS, and SSH). However, if you require other changes to the firewall, you will need to adjust them manually.

The process for this involves two steps:

1. Create a network security group.
2. Create an inbound rule allowing traffic on the ports you need.

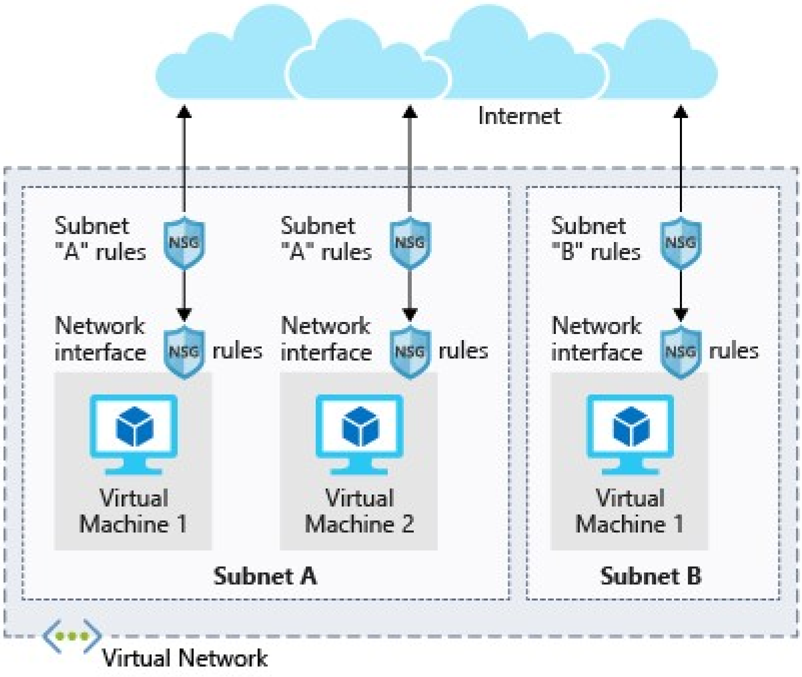
What is a network security group?

Virtual networks (VNets) are the foundation of the Azure networking model and provide isolation and protection. Network security groups (NSGs) are the primary tool you use to enforce and control network traffic rules at the networking level. NSGs are an optional security layer that provides a software firewall by filtering inbound and outbound traffic on the VNet.

Security groups can be associated to a network interface (for per host rules), a subnet in the virtual network (to apply to multiple resources), or both levels.

Security group rules

NSGs use *rules* to allow or deny traffic moving through the network. Each rule identifies the source and destination address (or range), protocol, port (or range), direction (inbound or outbound), a numeric priority, and whether to allow or deny the traffic that matches the rule.



Each security group has a set of default security rules to apply the default network rules described above. These default rules cannot be modified but *can* be overridden.

How Azure uses network rules

For inbound traffic, Azure processes the security group associated to the subnet and then the security group applied to the network interface. Outbound traffic is handled in the opposite order (the network interface first, followed by the subnet).

> ⚠ **Warning**
>
> Keep in mind that security groups are optional at both levels. If no security group is applied, then **all traffic is allowed** by Azure. If the VM has a public IP, this could be a serious risk, particularly if the OS doesn't provide a built-in firewall.

The rules are evaluated in *priority order*, starting with the **lowest priority** rule. Deny rules always **stop** the evaluation. For example, if a network interface rule blocks an outbound request, any rules applied to the subnet will not be checked. For traffic to be allowed through the security group, it must pass through *all* applied groups.

The last rule is always a **Deny All** rule. This is a default rule added to every security group for both inbound and outbound traffic with a priority of 65500. That means to have traffic pass through the security group, *you must have an allow rule*, or the final default rule will block it.

> ⓘ **Note**
>
> SMTP (port 25) is a special case. Depending on your subscription level and when your account was created, outbound SMTP traffic may be blocked. You can request to remove this restriction with business justification.

Creating network security groups

Security groups are managed resources like most everything in Azure. You can create them in the Azure portal or through command-line scripting tools. The challenge is in defining the rules. Let's look at defining a new rule to allow HTTP access and block everything else.

Next unit: Exercise - Configure network settings

Continue >

< Previous Unit 8 of 9 ⌄ Next >

✓ 100 XP ▶

Exercise - Configure network settings

10 minutes

This module requires a sandbox to complete. You have used 2 of 10 sandboxes for today. More sandboxes will be available tomorrow.

Activate sandbox

When we created the virtual machine (VM), we selected the inbound port *SSH* so we could connect to the VM. This created an NSG that's attached to the network interface of the VM. That NSG is blocking HTTP traffic. Let's update this NSG to allow inbound HTTP traffic on port 80.

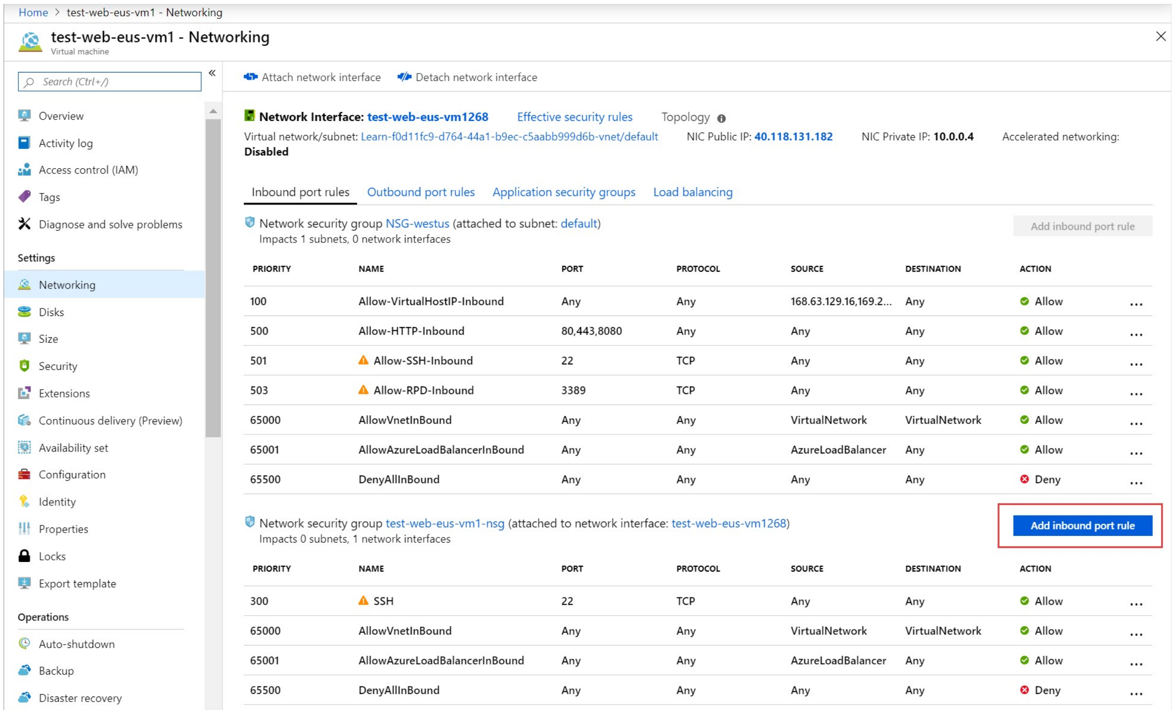
Update the NSG on the network interface

Port 80 is open on the NSG applied to the subnet. But port 80 is blocked by the NSG applied to the network interface. Let's fix that so we can connect to the website.

1. Switch back to the **Overview** panel for the virtual machine. You can find the VM under **All Resources**.

2. In the **Settings** section, select the **Networking** item.

3. You should see the NSG rules for the subnet in the top section and the NSG rules for the network interface in the bottom section of the same tab. In the bottom section, for the NSG rules for the network interface, select **Add inbound port rule**.



Network Interface: test-web-eus-vm1268

Virtual network/subnet: Learn-f0d11fc9-d764-44a1-b9ec-c5aabb999d6b-vnet/default NIC Public IP: 40.118.131.182 NIC Private IP: 10.0.0.4 Accelerated networking: Disabled

Network security group NSG-westus (attached to subnet: default)
Impacts 1 subnets, 0 network interfaces

PRIORITY	NAME	PORT	PROTOCOL	SOURCE	DESTINATION	ACTION	
100	Allow-VirtualHostIP-Inbound	Any	Any	168.63.129.16,169.2...	Any	✓ Allow	...
500	Allow-HTTP-Inbound	80,443,8080	Any	Any	Any	✓ Allow	...
501	⚠ Allow-SSH-Inbound	22	TCP	Any	Any	✓ Allow	...
503	⚠ Allow-RPD-Inbound	3389	TCP	Any	Any	✓ Allow	...
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	✓ Allow	...
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	✓ Allow	...
65500	DenyAllInBound	Any	Any	Any	Any	✗ Deny	...

Network security group test-web-eus-vm1-nsg (attached to network interface: test-web-eus-vm1268)
Impacts 0 subnets, 1 network interfaces

PRIORITY	NAME	PORT	PROTOCOL	SOURCE	DESTINATION	ACTION	
300	⚠ SSH	22	TCP	Any	Any	✓ Allow	...
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	✓ Allow	...
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	✓ Allow	...
65500	DenyAllInBound	Any	Any	Any	Any	✗ Deny	...

4. Switch to the **Basic** mode.

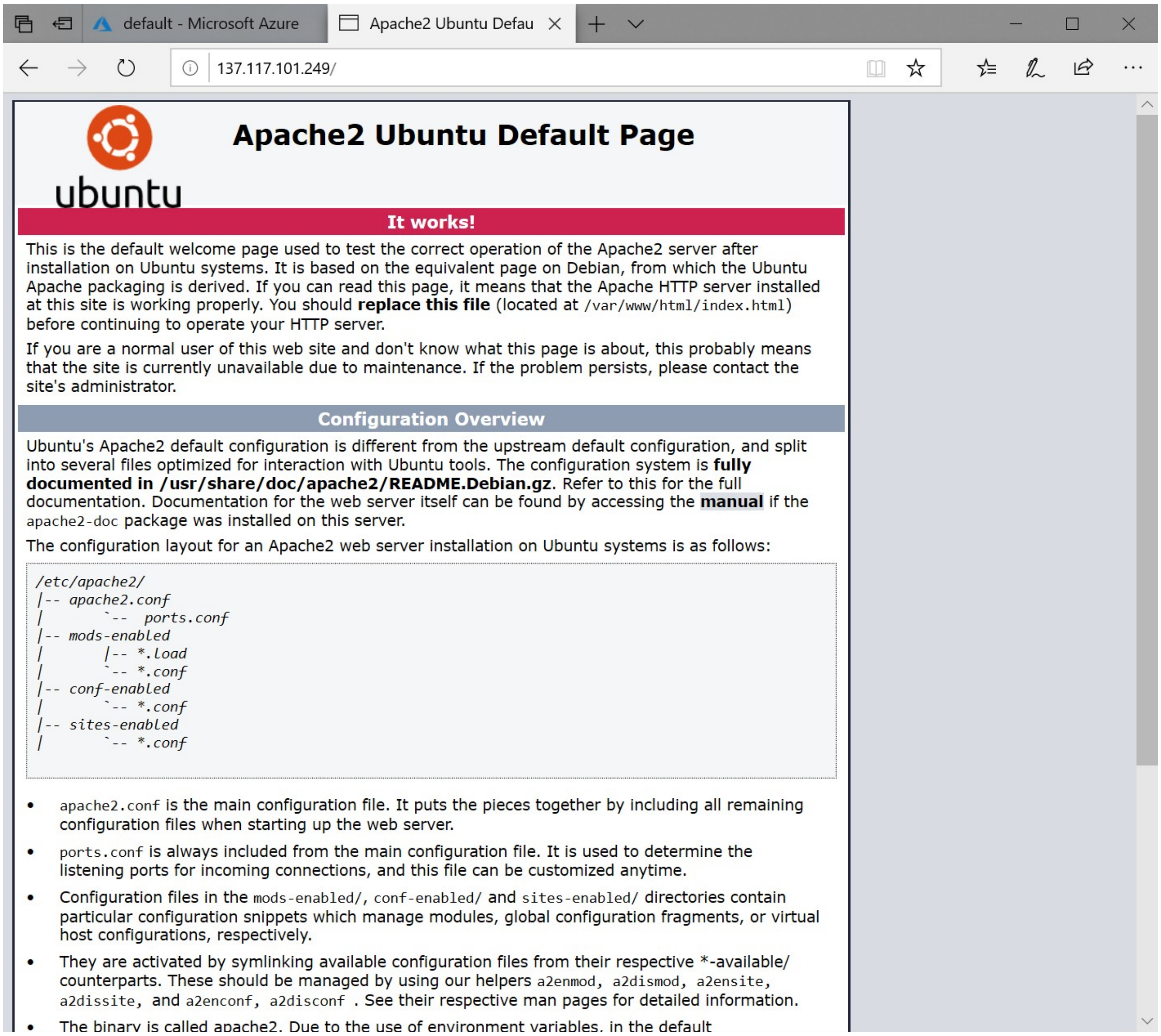
5. Add the information for our HTTP rule:

    - Set the **Service** to be HTTP. This sets up your port range.
    - Set the **Priority** to **310**.
    - Give the rule a name; use **allow-http-traffic**.
    - Give the rule a description.

6. Click **Add** to create the rule.

Open the default webpage

Use the IP address of the server to make an HTTP request. It should now work.



Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective *-available/ counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf` . See their respective man pages for detailed information.
- The binary is called apache2. Due to the use of environment variables, in the default

One more thing

Always make sure to lock down ports used for administrative access. An even better approach is to create a VPN to link the virtual network to your private network and only allow RDP or SSH requests from that address range. You can also change the port used by SSH to be something other than the default. Keep in mind that changing ports is not sufficient to stop attacks. It simply makes it a little harder to discover.

Next unit: Summary

Continue >

Summary

3 minutes

In this module, you learned how to create a Linux VM using the Azure portal. You then connected to the public IP address of the VM and managed it with an SSH connection.

You learned that while SSH allows us to interact with the operating system and software of the virtual machine, the portal will enable us to configure the virtual hardware and connectivity. We also could have used PowerShell or the Azure CLI, if a command-line or scriptable environment were preferred.

Clean up

The sandbox automatically cleans up your resources when you're finished with this module.

When you're working in your own subscription, it's a good idea at the end of a project to identify whether you still need the resources you created. Resources left running can cost you money. You can delete resources individually or delete the resource group to delete the entire set of resources.

Check your knowledge

1. True or false: for security reasons, you must use an image from the official Azure Marketplace when creating a new virtual machine.

○ True

● False ✓

Azure lets you configure your virtual machines to meet your needs. This includes support for using your own VM images.

2. What is the effect of the default network security settings for a new virtual machine?

○ Neither outbound nor inbound requests are allowed.

◉ Outbound request are allowed. Inbound traffic is only allowed from within the virtual network. ✓

Outbound requests are considered low risk, so they are allowed by default. Inbound traffic from within the virtual network is allowed. By placing a VM in a virtual network, the VM owner is implicitly opting-in to communication among the resources in the virtual network.

○ There are no restrictions: all outbound and inbound requests are allowed.

3. Suppose you have several Linux virtual machines hosted in Azure. You will administer these VMs remotely over SSH from three dedicated machines in your corporate headquarters. Which of the following authentication methods would typically be considered best-practice for this situation?

○ Username and password

○ Private key

◉ Private key with passphrase ✓

Private key access with a passphrase is the most secure option. Even if an attacker acquires your private key, they will be unable to use it without the passphrase.

Module complete:

Unlock achievement