School of Information Studies

Syracuse University

IST 615 – Cloud Management

Lab #1
Virtual Machines in Azure

IMPORTANT INSTRUCTIONS[1]:

i) Read this document in its entirety BEFORE starting the lab.

ii) You will need to access your Visual Studio Enterprise Subscription to access the Microsoft Azure Cloud resources needed to complete this lab

iii) Some screenshots may not appear "exactly" as depicted in this document. They are intended as a guide and you should adapt to your own local situation/environment.

iv) You will need to provide screen captures that indicate your progress with the steps required in this lab. There will be notices for you to capture screenshots during some of the steps in this lab. Make sure each capture has been correctly completed in your system.

Note: This lab's content is based on material from Microsoft. The content has been adapted for clarity and to fit the objectives of the lab.

[1] School of Information Studies, Syracuse University
Lab created by Carlos E. Caicedo
Last updated on September/2024 by Carlos Caicedo

Contents

1 Introduction

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 various cloud-based infrastructure that uses 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. The company wants to keep their existing website, which was rewritten last year, but have it operate in a cloud-based environment.

The website is running on Apache with Linux, and because it's already up and running, you decide to move it directly into Azure by using 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

2 Create a Linux virtual Machine in Azure

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.

2.1 Introduction to Azure Virtual Machines

Azure Virtual Machines is an on-demand, scalable cloud-computing resource. Each virtual machine includes processor, 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) session to connect directly to a running VM and execute commands as if you were on a local computer.

2.1.1 Run 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.

2.2 Create 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.

2.3 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.

- 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.

2.4 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.

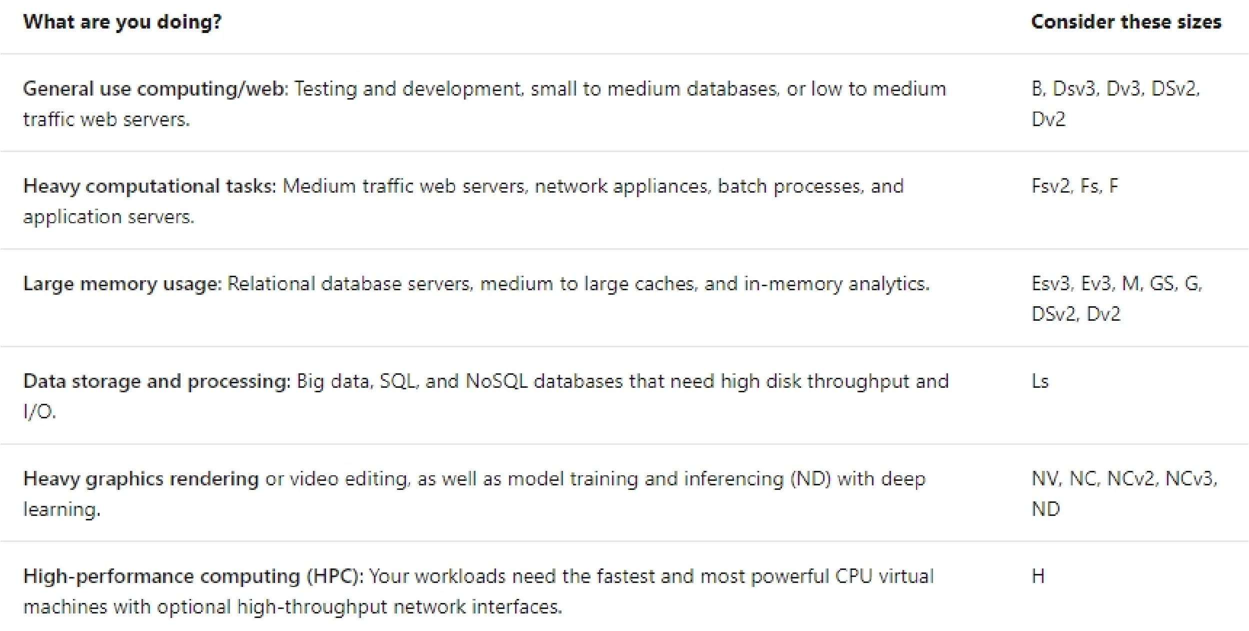
Anything that a computer can have installed can be included in an image. You can create a VM from an image that's preconfigured to precisely handle the tasks you need, such as hosting a web app using the Apache HTTP Server software.

2.5 Size 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.

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

2.6 Choose 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.

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.

2.6.1 Map 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. 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.

2.6.1.1 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.

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.

2.6.2 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. We recommend that you shut down the VM to clear out any processes that are in progress. 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.

2.7 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.

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.

3 Exercise - Decide an authentication method for SSH

Before we can create a Linux virtual machine in Azure, we will need to think about remote access. We want to be able to sign in to our Linux web server to configure the software and perform maintenance. The default approach to administering Linux VMs hosted in Azure is SSH.

3.1 What is SSH?

Secure Shell (SSH) is an encrypted connection protocol that allows secure sign-ins over unsecured connections. SSH allows you to connect to a terminal shell from a remote location using a network connection.

There are two approaches we can use to authenticate an SSH connection: **username and password**, or an **SSH key pair**.

Although SSH provides an encrypted connection, using passwords with SSH connections leaves the VM vulnerable to brute-force attacks of passwords. A more secure and preferred method of connecting to a Linux VM with SSH is a public-private key pair, also known as SSH keys.

With an SSH key pair, you can sign in to Linux-based Azure virtual machines without a password. This is a more secure approach if you only plan to sign in to the VM from a few computers. If you need to be able to access the Linux VM from a variety of locations, a username and password combination might be a better approach. There are two parts to an SSH key pair: a public key and a private key.

- The **public key** is placed on your Linux VM or any other service that you wish to use with public-key cryptography. This can be shared with anyone.

- The **private key** is what you present to verify your identity to your Linux VM when you make an SSH connection. Consider this confidential information and protect this like you would a password or any other private data.

You can use the same single public-private key pair to access multiple Azure VMs and services.

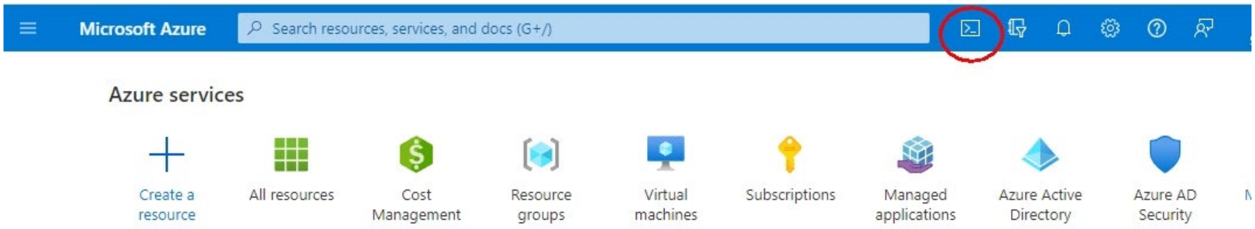
3.2 Create the SSH key pair

On Windows 10, Windows 11, Linux, and macOS, you can use the built-in `ssh-keygen` command to generate the SSH public and private key files.

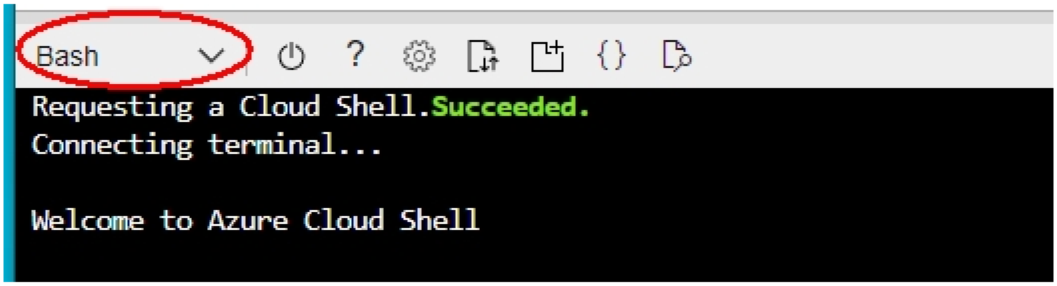
Windows 11 includes an SSH client. Very early versions of Windows 10 require additional software to use SSH; check the documentation for full details. Alternatively, you can install the Linux subsystem for Windows (on a Windows machine) and get the same functionality.

We will use Azure Cloud Shell, which stores the generated keys in Azure in your private storage account. You can also type these commands directly into your local shell if you prefer. You will need to adjust the instructions throughout this module to reflect a local session if you take this approach.

1. Sign in to the Azure portal with your Visual Studio Enterprise Subscription
2. Activate the Cloud Shell by clicking on the icon for it (near the top right of the portal window)



Important: Make sure the BASH version of the Cloud Shell is active, not the PowerShell version.



```
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell
```

Here is the minimum command necessary to generate the key pair for an Azure VM. This creates an SSH protocol 2 (SSH-2) RSA public-private key pair. The minimum length is 2048, but for the sake of this learning module we will use 4096.

1. Type the following command in Cloud Shell.

   ```
   ssh-keygen -m PEM -t rsa -b 4096 -f ~/lab01key
   ```

2. Press `Enter` . The command creates two files in you home directory: `lab01key` and `lab01key.pub` . The files are overwritten if they exist.
3. Enter a passphrase that you'll remember. You'll need this passphrase when you use the SSH key to access the VM. (suggestion: use *lab01* as the passphrase)

3.2.1 Private key passphrase

You can provide a passphrase while generating your private key. This is a password you must enter when you use the key. This passphrase is used to access the private SSH key file and is not the user account password.

When you add a passphrase to your SSH key, it encrypts the private key using 128-bit AES so that the private key is useless without the passphrase to decrypt it.

We strongly recommended that you add a passphrase. If an attacker stole your private key and that key did not have a passphrase, they would be able to use that private key to log in to any servers that have the corresponding public key. If a passphrase protects a private key, it cannot be used by that attacker. This provides an additional layer of security for your infrastructure on Azure.

3.3 Use the SSH key pair with an Azure Linux VM

After you have the key pair generated, you can use it with a Linux VM in Azure. You can supply the public key during the VM creation, or add it after the VM has been created.

You can view the contents of the file in Cloud Shell by running the following command.

```
cat ~/lab01key.pub
```

It will look something like the following output:

```
ssh-rsa
XXXXXXXXXXc2EAAAADAXABAAABAXC5Am7+fGZ+5zXBGgXS6GUvmsXCLGc7tX7/rViXk3+eShZzaXnt75gUmT1
I2f75zFn2hlAIDGKWf4g12KWcZxy81TniUOTjUsVlwDDmXUXxESL/UfJKfbdstBhTOdy5EG9rYWA0K43SJmwP
hH28BpoLfXXXXXGX/ilsXXXXXKgRLiJ2W19MzXHp8z3Lxw7r9wx3HaVlP4XiFv9U4hGcp8RMI1MP1nNesFlOB
pG4pV2bJRBTXNXeY4l6F8WZ3C4kuf8XxOo09mXaTpvZ3T1841altmNTZCcPkXuMrBjYSJbA8npoXAXNwiivyo
e3X2KMXXXXXdXXXXXXXXXXCXXXXX/ somename@someserver
```

In a later step in this lab, you will need to select the output that appeared in your cloud shell session and copy it so you can use it. The copy should be to your computer’s memory/clipboard. You could also later paste the copy of your public key into a simple text editor (e.g. Notepad) to reuse it later but be VERY CAREFUL that the editor does NOT introduce any line breaks to the key, otherwise it will become useless. In other words, you might need to re-execute the *cat* command described previously to get the public key output.

3.3.1 Use the SSH key when creating a Linux VM

To apply the SSH key while creating a new Linux VM, you will need to copy the contents of the public key (as indicated previously) and supply it to the Azure portal, *or* supply the public key file to the Azure CLI or Azure PowerShell command. We'll use the first approach when we create our Linux VM.

Now that we have our public key, let's switch to the Azure portal and create a Linux VM.

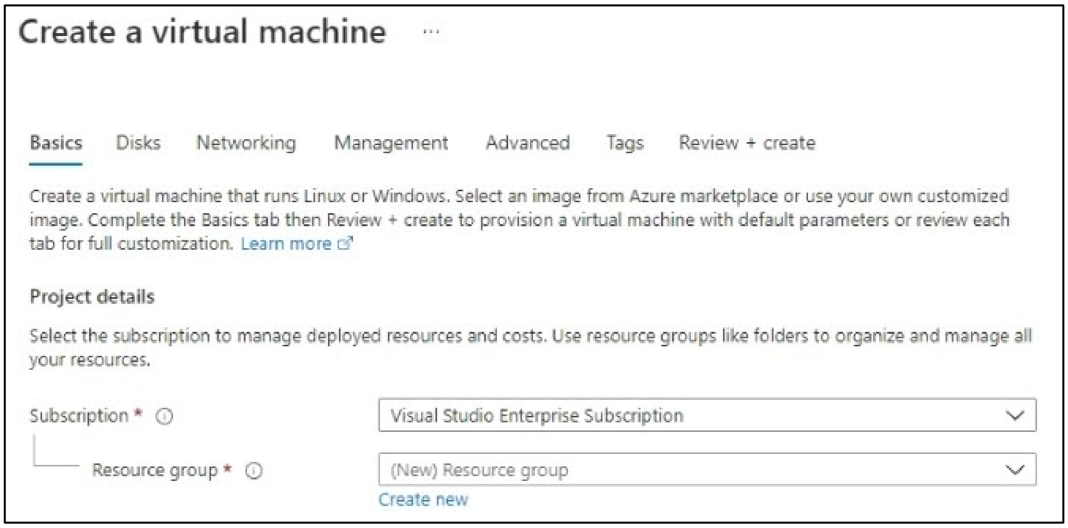
4 Exercise - Create a Linux virtual machine with the Azure portal

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

4.1 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. On the Azure portal menu or from the **Home** page, under **Azure services**, select **Virtual machines**. Alternatively, you can enter *Virtual machines* in the top search box, and press `Enter`. The **Virtual machines** pane appears.
2. In the top menu bar, select **Create > Azure virtual machine**. The **Create a virtual machine** pane appears.


Create a virtual machine ···

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. Learn more

Project details

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

Subscription * : Visual Studio Enterprise Subscription

Resource group * : (New) Resource group

4.2 Configure the VM settings and configure the network

The VM creation experience in the portal is presented in a wizard format to walk you through all the configuration areas for the VM. Selecting **Next** takes you to the next configurable tab. However, you can move between the tabs at will by selecting them. After you complete all the required options (identified with red asterisks), you can skip the remainder of the wizard experience, and start creating the VM by selecting **Review + create** at the bottom of the wizard. However, for our purposes, to start creating a VM, follow the steps mentioned below but please note that you should ONLY change the values of each of the Settings mentioned in this document to the value presented in the following steps. If a setting's value is not mentioned and/or described, leave its current default value as defined by Azure. In other words, **do not change a setting's value unless instructed to do so**:

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

Setting	**Value**
Project details	
Subscription	Visual Studio Enterprise Subscription
Resource group	Select ***Create new*** and type *IST615lab01* in the field
Instance details	
Virtual machine name	Enter *lab01vm* as the name of the Linux VM that you will create. VM names must be between 1 and 64 characters and be comprised of numbers, letters, and dashes.
Region	Select ***(US) East US***
Availability options	Select **No infrastructure redundancy required**. Availability options 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 lab we will not need this service.
Image	From the dropdown list, select **Ubuntu Server 22.04 LTS – x64 Gen2**
VM architecture	**x64** should be selected
Run with Azure Spot discount	Leave unselected
Size	**Standard_D2s_v3**. This option gives you two vCPUs with 8 GB of RAM.
Administrator account	
Authentication type	SSH public key
Username	*azureuser* is the default value, leave it.
SSH public key source	Select: **Use existing public key**
SSH public key	Paste the SSH key from the public key file you created in the previous section. It should look similar to the example shown in section 3 with no additional whitespace or line-feed characters.
Inbound port rules	Allow selected ports
Select inbound ports	SSH (22)

2. Select **Next: Disks** to open the **Disks** tab.
3. On the **Disks** pane, enter the following values for each setting.

Setting	Value
Disk options	
OS disk type	Premium SSD (locally-redundant storage)
Key management	Platform-managed key

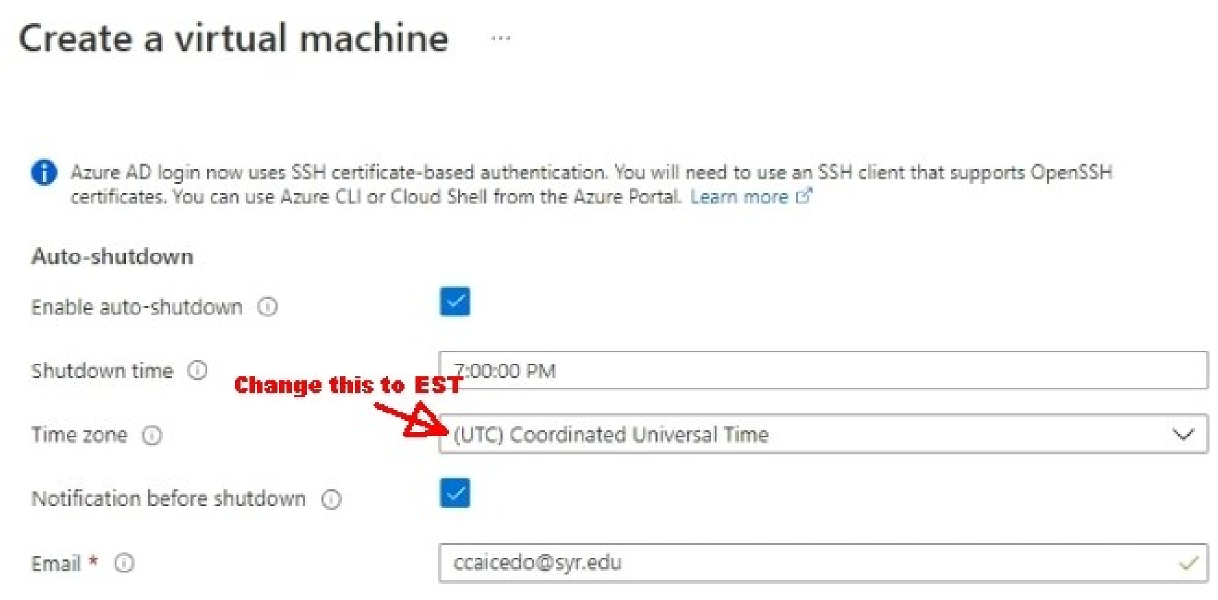
4. Select **Next: Networking** to move to the **Networking** tab.
5. On the **Networking** pane, accept all the default values for each setting.

   In a production environment where we already have other components, you'd want to use an *existing* virtual network. That way, your VM can communicate with the other cloud services in your solution. If there wasn't one defined in this location yet, you could create it here and configure the:

   - **Subnet**: The first subnet to subdivide the address space - it must fit within the defined address space. After the VNet is created, you could add additional subnets.
   - **Public IP**: The overall IPv4 space available to this network.

   By default, Azure creates 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.

6. Select **Next: Management** to move to the **Management** tab.
7. Scroll down to the **Auto-shutdown** section. This section will allow you to define a time of the day in which your VM will automatically shutdown in case you have forgotten to shut it down yourself. (When the VM is shutdown it stops most of the charges related to the operations of the VM, except for storage). You should set a Shutdown time and a Time zone that fits your needs (e.g. 11 p.m. , Eastern Time). See the figure in the next page for an example. Don’t change any of the other parameters.



Create a virtual machine

Azure AD login now uses SSH certificate-based authentication. You will need to use an SSH client that supports OpenSSH certificates. You can use Azure CLI or Cloud Shell from the Azure Portal. Learn more

Auto-shutdown

Enable auto-shutdown ☑

Shutdown time: 7:00:00 PM

Time zone: (UTC) Coordinated Universal Time

Change this to EST

Notification before shutdown ☑

Email *: ccaicedo@syr.edu

8. The rest of the options in the wizard 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.
9. Finish configuring the VM and creating the image by selecting **Review + create**.
10. After the system validates your options, and gives you details about the VM being created, select **Create** to create and deploy the VM. The Azure dashboard will show the VM that's being deployed. This may take several minutes.

While it's deploying, let's consider what we can do with this VM.

SCS01- Capture a screenshot that shows that the VM has finished creating and has a public and private IP address. Include this screenshot in your lab report.

5 Azure virtual machines IP addresses and SSH options

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?

5.1 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 that 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 previously described, we will use the public IP address of the VM in this lab.

5.2 Connect to the VM with SSH

To connect to the VM via SSH, you need the following items:

- Public IP address of the VM
- Username of the local account on the VM
- 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 section, you'll use this information to open a secure terminal on the VM using SSH.

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

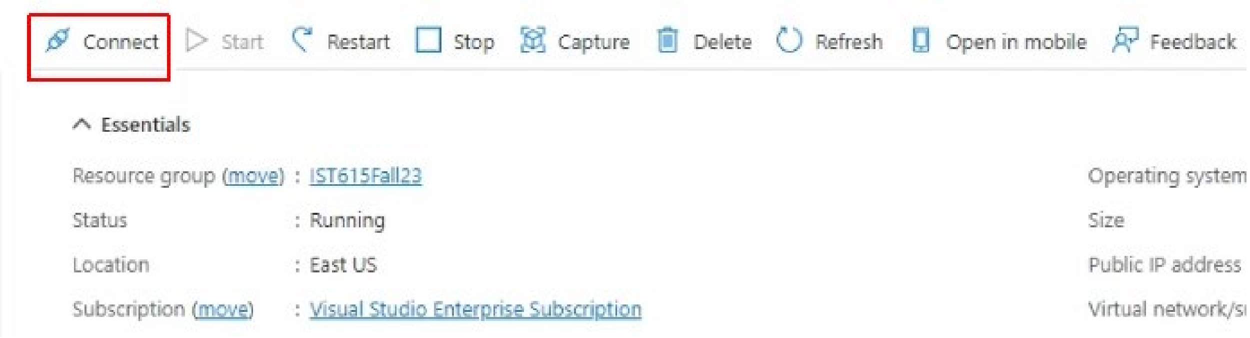
Next, let's connect to the VM using SSH.

6 Exercise - Connect to a Linux virtual machine with SSH

Let's connect to our Linux VM with SSH, and configure Apache, so we have a running web server.

6.1 Get the public IP address of the VM

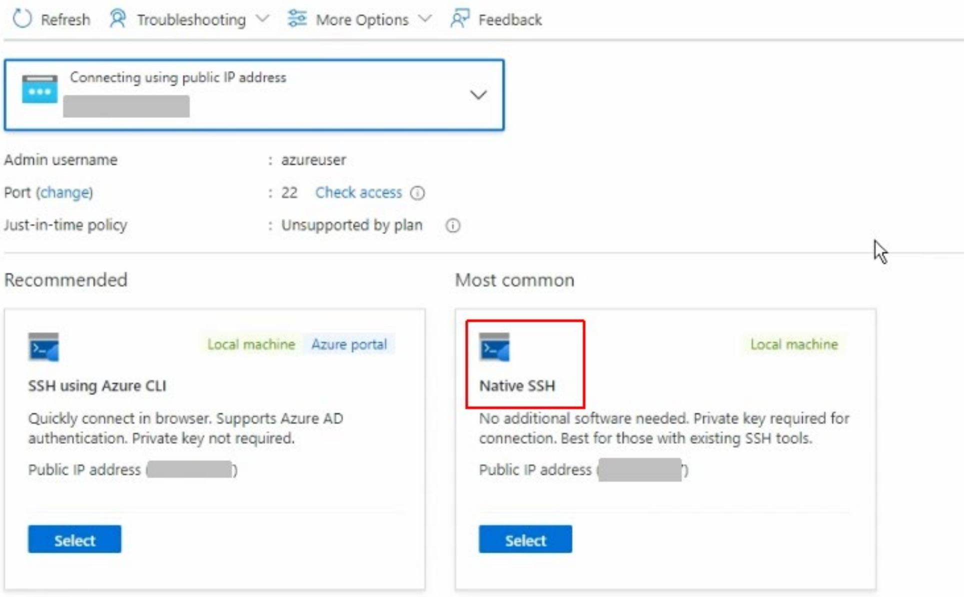
1. In the Azure portal from the previous exercise, select **Go to resource**. The **Overview** pane for the virtual machine that you just created appears. Alternatively, you can find the VM under **All Resources** if you need to open it. The overview pane enables you to:
   - See if the VM is running
   - Stop or restart the VM
   - Get the public IP address of the VM
   - See the activity of the CPU, disk, and network
2. Select **Connect** -> Connect



Resource group (move) : IST615Fall23
Status : Running
Location : East US
Subscription (move) : Visual Studio Enterprise Subscription

Operating system
Size
Public IP address
Virtual network/s

3. You should now see a list of “Recommended” and “Most Common” options for connecting to the VM. Choose the “Native SSH” option as shown in the next figure. DO NOT choose any other option.



Connecting using public IP address
[redacted]

Admin username : azureuser
Port (change) : 22 Check access
Just-in-time policy : Unsupported by plan

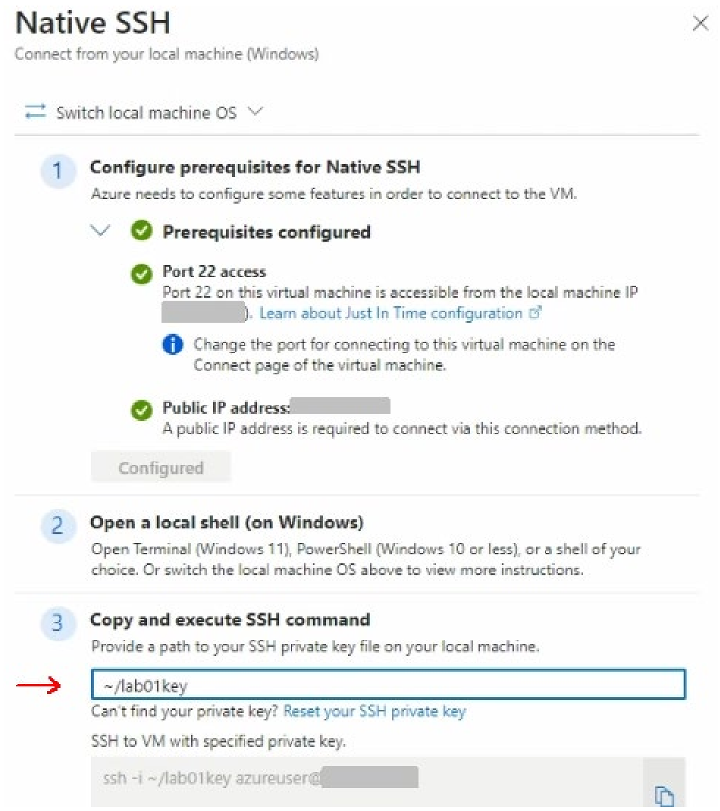
Recommended

Local machine Azure portal
SSH using Azure CLI
Quickly connect in browser. Supports Azure AD authentication. Private key not required.
Public IP address ([redacted])

Most common

Local machine
Native SSH
No additional software needed. Private key required for connection. Best for those with existing SSH tools.
Public IP address ([redacted])

4. Under item 3 in the *Native SSH* sub-window, type the path and name of your private key file. in the “Private key path” field. If you followed all the steps in section 3 of this lab correctly, you should type: **`~/lab01key`**



Native SSH

Connect from your local machine (Windows)

1 **Configure prerequisites for Native SSH**

Azure needs to configure some features in order to connect to the VM.

Prerequisites configured

Port 22 access

Port 22 on this virtual machine is accessible from the local machine IP [redacted]). Learn about Just In Time configuration

Change the port for connecting to this virtual machine on the Connect page of the virtual machine.

Public IP address: [redacted]

A public IP address is required to connect via this connection method.

2 **Open a local shell (on Windows)**

Open Terminal (Windows 11), PowerShell (Windows 10 or less), or a shell of your choice. Or switch the local machine OS above to view more instructions.

3 **Copy and execute SSH command**

Provide a path to your SSH private key file on your local machine.

~/lab01key

Can't find your private key? Reset your SSH private key

SSH to VM with specified private key.

```
ssh -i ~/lab01key azureuser@[redacted]
```

5. Copy the command shown in the gray box (of item 3) to your clipboard

6.2 Connect with SSH

1. Paste the command from your clipboard into your **Azure Cloud Shell**. It should look something like the following sample but with a different IP address

```
ssh -i ~/lab01key azureuser@137.117.101.249
```

2. The first time we connect, SSH will ask us about authenticating against an unknown host. SSH is informing you that you've never connected to this server before. If that's true, it's perfectly normal, and you can respond with **yes** to save the fingerprint of the server in the known host file.

```
The authenticity of host '137.117.101.249 (137.117.101.249)' can't be
established.
ECDSA key fingerprint is
SHA256:w1h08h4ie1iMq7ibIVSQM/PhcXFV7O7EEhjEqhPYMWY.
```

```
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '137.117.101.249' (ECDSA) to the list of known
hosts.
```

3. Enter the passphrase you used when you created the SSH key pair.
4. In the shell command prompt for Linux, try executing a few Linux commands:
    - `ls -la /`: Shows the root of the disk
    - `ps -l`: Shows all the running processes
    - `dmesg`: Lists all the kernel messages
    - `lsblk`: Lists all the block devices - here you will see your drives

SCS02 - Capture a screenshot that shows that you have connected to the VM. Include this screenshot in your lab report.

6.3 Install software onto the VM

As you can see, SSH enables you to work with the Linux VM just like a local computer. You can administer this VM as you would any other Linux computer: installing software, configuring roles, adjusting features, and other everyday tasks. Let's focus on installing software for a moment.

You can also install software from the internet when you are connected to the VM via SSH. Azure machines are, by default, internet connected. You can use standard commands to install popular software packages directly from standard repositories. Let's use this approach to install Apache.

6.3.1 Install the Apache web server

Apache is available within Ubuntu's default software repositories, so we will install it using conventional package management tools:

1. Start by updating the local package index to reflect the latest upstream changes.

    ```
    sudo apt-get update
    ```

2. Next, install Apache.

    ```
    sudo apt-get install apache2 -y
    ```

3. It will take a few minutes for all the apache2 and related software packages to be installed in your system

4. After all the packages are installed, the web server capability (service) on your VM should start automatically - we can check the status using the `systemctl` command.

   ```
   sudo systemctl status apache2 --no-pager
   ```

   The `systemctl` command output should indicate that the HTTP server is “Active”.

SCS03 - Capture a screenshot that shows the output of the systemctl command and that you have an active HTTP server.

5. Finally, we can try retrieving the default page through the public IP address of the VM. However, even though the web server is running on the VM, you won't get a valid connection or response. Do you know why?

We need to perform one more step to be able to interact with the web server. Our virtual network is blocking the inbound request. We can change that through configuration. Let's look at allowing the inbound request next.

7 Network and security settings

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.

Let's explore the Azure network configuration to see how to use the built-in security support to harden our server.

7.1 Open 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 (for example, other resources on the same local network) and from Azure Load Balancer (probe checks).

There are two steps for 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:

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

7.1.1 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.

7.1.1.1 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 previously described. These default rules cannot be modified but *can* be overridden.

7.1.1.2 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.

7.2 Create 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.

8 Exercise - Configure network settings

When we created the VM, we selected the inbound port *SSH* (port 22) 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 thus, no web traffic (web pages) is allowed into our VM. We need to correct this. Let's update this NSG to allow inbound HTTP traffic on port 80.

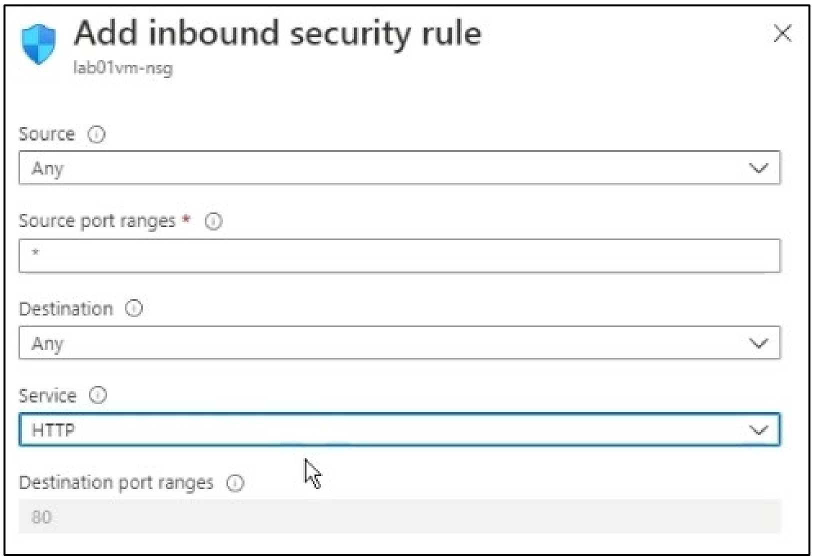
8.1 Update the NSG on the network interface

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 have the VM act as a web server.

1. Switch back to the **Overview** pane for the VM. You can find the VM by going to *Home* and selecting *Virtual Machines* (or by going to the *All Resources* option).
2. Select your VM. In the left menu pane, under **Networking** select **Network settings**.
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 **+Create port rule** and then select **Inbound port rule**.

The **Add inbound security rule** pane appears.



Add inbound security rule

lab01vm-nsg

Source: Any

Source port ranges *: *

Destination: Any

Service: HTTP

Destination port ranges: 80

4. Enter the following values for our HTTP rule.

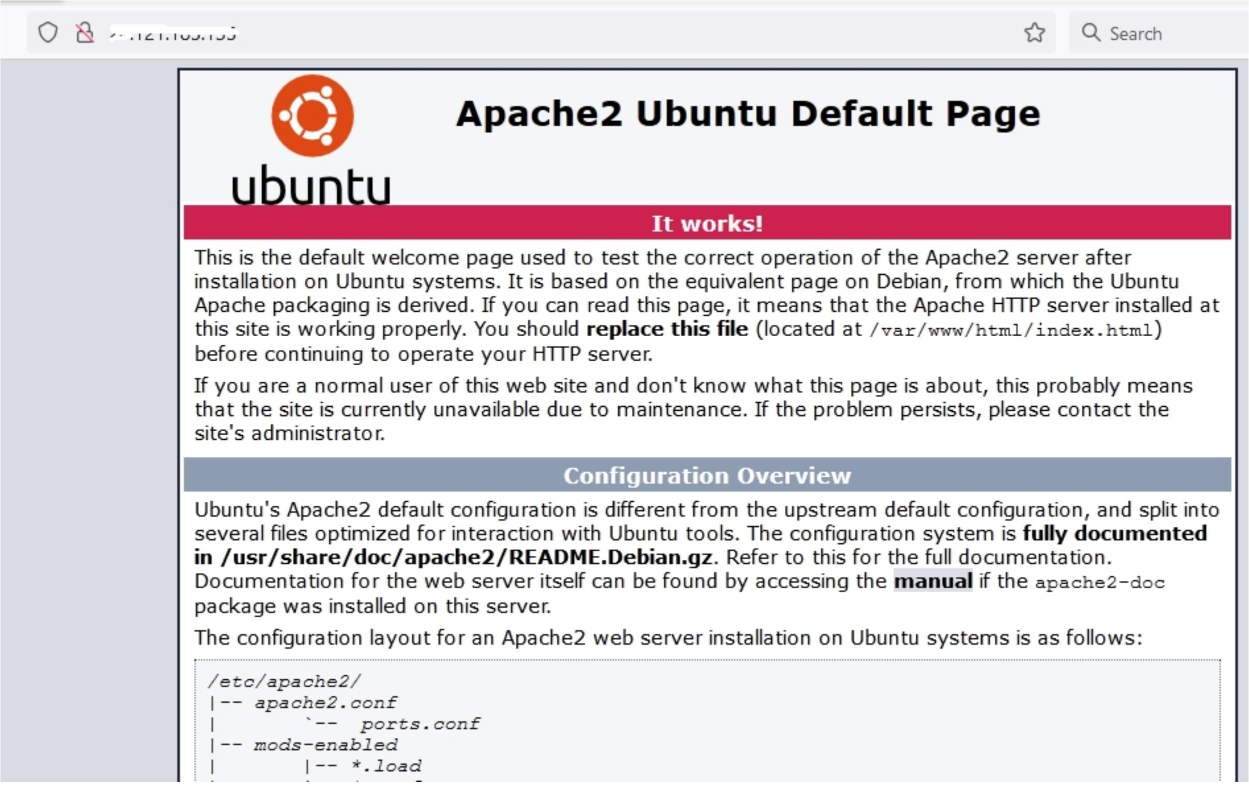
Setting	Value
Service	HTTP
Priority	310
Name	allow-http-traffic
Description	Allows http traffic

5. Select **Add** to create the rule. The **Networking** pane for your VM reappears and will have the updated rule almost immediately but it could take a few seconds for it to display.

SCS04 - Capture a screenshot that shows that you have added the new inbound port rule to the VM. Include this screenshot in your lab report.

8.2 Open the default webpage

To make an HTTP request, copy and paste the **Public IP address** of your VM (it is now a web server) into a browser, and press `Enter`. You should see your web server working and displaying a web page similar to the one shown in the following figure.



Apache2 Ubuntu Default Page

ubuntu

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
```

SCS05 - Capture a screenshot that shows that the Apache server default web page loads. The IP address of your VM/website should be visible. Include this screenshot in your lab report.

9 Summary

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 was preferred.

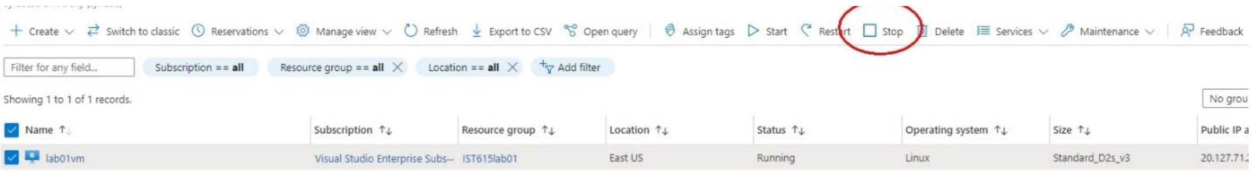
10 Stopping your VM

Resources in the cloud left running/active and not being used can cost you money. You can stop or pause these resources so that charges are minimized. Let’s proceed to stop your VM so that you are not charged for its activity. We will re-start the VM at a later date without having to go through the creation steps previously covered in this lab.

Please note that even after stopping the VM, you will continue to be charged for other resources related to it such as storage and networking.

To stop the VM carry out the following steps:

1. Switch back to the **Overview** pane for the VM. You can find the VM under **All Resources**.
2. Proceed to select your VM and press the **Stop** button in the pane.



Showing 1 to 1 of 1 records.

Name	Subscription	Resource group	Location	Status	Operating system	Size	Public IP a
lab01vm	Visual Studio Enterprise Subs···	IST615lab01	East US	Running	Linux	Standard_D2s_v3	20.127.71.

3. Once the VM is stopped you can close your Azure portal session. Your VM will be available for you to use in a later session if needed.

11 Lab Report

For your report, please include the following items (Remember to include a title page):

Part 1 (50 points):

Include all the screenshots required throughout the lab guide. Provide a small description of what is being shown in each screenshot

Part 2 (50 points):

You accessed your VM instance using SSH. Use that experience (and some additional research), to answer the following questions.

1. What is a key pair and what is it used for?
2. Who stores the public portion of the key pair? Who stores the private portion of the key pair?
3. What is SSH? What is it used for?
4. When you make a change to a network security group rule, does the change affect only the instance you're currently working on or other instances, too? Explain.
5. What is the effect of the default network security settings for a new virtual machine?
   a) Neither outbound nor inbound requests are allowed.

   b) Outbound request are allowed. Inbound traffic is only allowed from within the virtual network.
   c) There are no restrictions: all outbound and inbound requests are allowed.

6. 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?

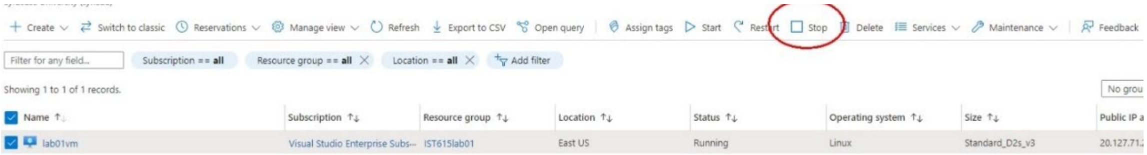
   a) Username and password
   b) Private key
   c) Private key with passphrase

12 Appendix. Deleting all VM related resources

FOLLOW THESE INSTRUCTIONS ONLY IF YOU NEED TO CLEAN UP ALL RESOURCES CONFIGURED IN THIS LAB EITHER BECAUSE YOU NEED TO RESTART THE LAB AGAIN OR BECAUSE YOU WON'T NEED THE RESOURCES IN THE FUTURE.

You can delete resources individually or delete the resource group to delete the entire set of resources. Deleting the resource group is the best way to remove all resources associated with your VM so that you are not continuously charged for them. To delete all the VM resources follow the next steps:

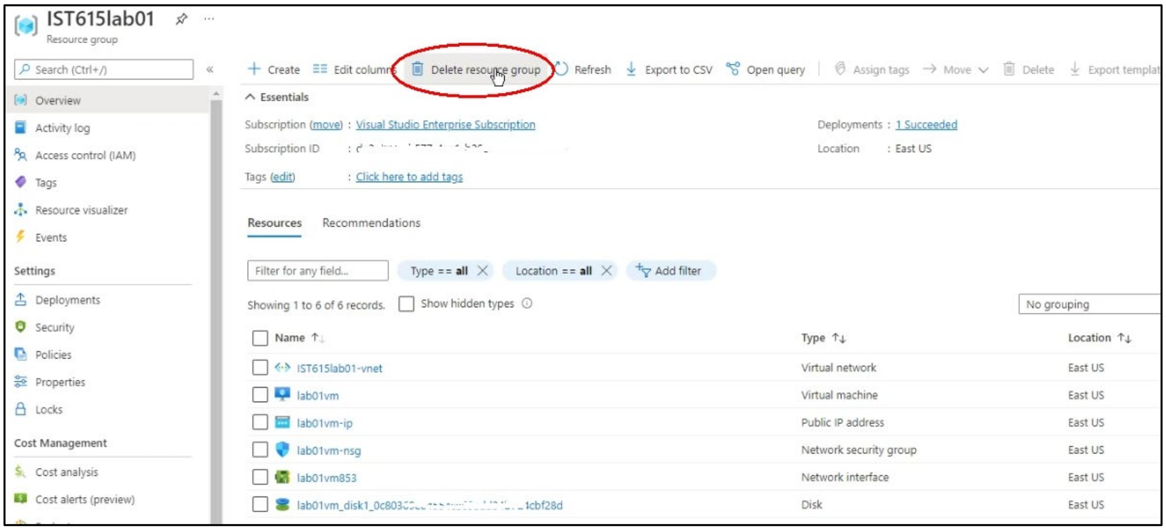
1. Switch back to the **Overview** pane for the VM. You can find the VM under **All Resources**.
2. Before deleting your VM it is good practice to first shut it down (stop its operation). Proceed to select your VM and press the **Stop** button in the pane.



Showing 1 to 1 of 1 records.

☑ Name	Subscription	Resource group	Location	Status	Operating system	Size	Public IP a
☑ lab01vm	Visual Studio Enterprise Subs...	IST615lab01	East US	Running	Linux	Standard_D2s_v3	20.127.71.

3. Once the VM is stopped, navigate to **Home -> Resource groups** or type *Resource groups* in the search bar of the portal.
4. Select the *IST651lab01* resource group and click on it
5. Click on *Delete resource group* to start the deletion of all resources associated with the VM created in this lab. Once the process finishes, the resource group will not be listed anymore and you will not continue to be charged for any resources that were previously present in the group.



Subscription (move) : Visual Studio Enterprise Subscription
Subscription ID : [redacted]
Tags (edit) : Click here to add tags
Deployments : 1 Succeeded
Location : East US

Showing 1 to 6 of 6 records.

□ Name	Type	Location
□ IST615lab01-vnet	Virtual network	East US
□ lab01vm	Virtual machine	East US
□ lab01vm-ip	Public IP address	East US
□ lab01vm-nsg	Network security group	East US
□ lab01vm853	Network interface	East US
□ lab01vm_disk1_0c803[redacted]1cbf28d	Disk	East US