Contents

[VIRTUAL MACHINE 2](#_Toc124705986)

[WHY TO USE VIRTUAL MACHINES 2](#_Toc124705987)

[AZURE VIRTUAL MACHINE SERVICE 3](#_Toc124705988)

[AZURE VIRTUAL MACHINE TYPES 3](#_Toc124705989)

[AZURE VIRTUAL MACHINE – KEY CONCEPTS 4](#_Toc124705990)

[CREATING/DEPLOYING A VIRTUAL MACHINE 4](#_Toc124705991)

[VIRTUAL MACHINE DISK 16](#_Toc124705992)

[STATE OF VM 29](#_Toc124705993)

[AVAILABILITY 29](#_Toc124705994)

[AVAILABILITY SETS 29](#_Toc124705995)

[AVAILABILITY ZONES 33](#_Toc124705996)

[SCALE VMS IN AZURE 35](#_Toc124705997)

[WHAT IS AZURE BATCH? 36](#_Toc124705998)

[VIRTUAL MACHINE SCALE SETS 36](#_Toc124705999)

[VM -BOOT DIAGNOSTICS 44](#_Toc124706000)

[VM- SERIAL CONSOLE 44](#_Toc124706001)

[VM – RUN COMMAND 44](#_Toc124706002)

[AZURE DEDICATED HOST 44](#_Toc124706003)

[ADVANTANGE OF DEDICATED HOST 44](#_Toc124706004)

[CREATING DEDICATED HOST 44](#_Toc124706005)

[AZURE BASTION SERVICE 45](#_Toc124706006)

[USE CASE 45](#_Toc124706007)

[SETTING UP BASTION SERVICE 47](#_Toc124706008)

[VIRTUAL MACHINE - CUSTOM IMAGES 50](#_Toc124706009)

[TYPES OF CUSTOM IMAGES 51](#_Toc124706010)

[CREATING A SPECIALIZED IMAGE 51](#_Toc124706011)

# VIRTUAL MACHINE

* A virtual machine is a virtual emulation of physical computer system. It is a virtual environment that includes components such as CPUs, memory , network interfaces and mofe giving the end users the same experiece in VM as they would have on a normal physical computer machine

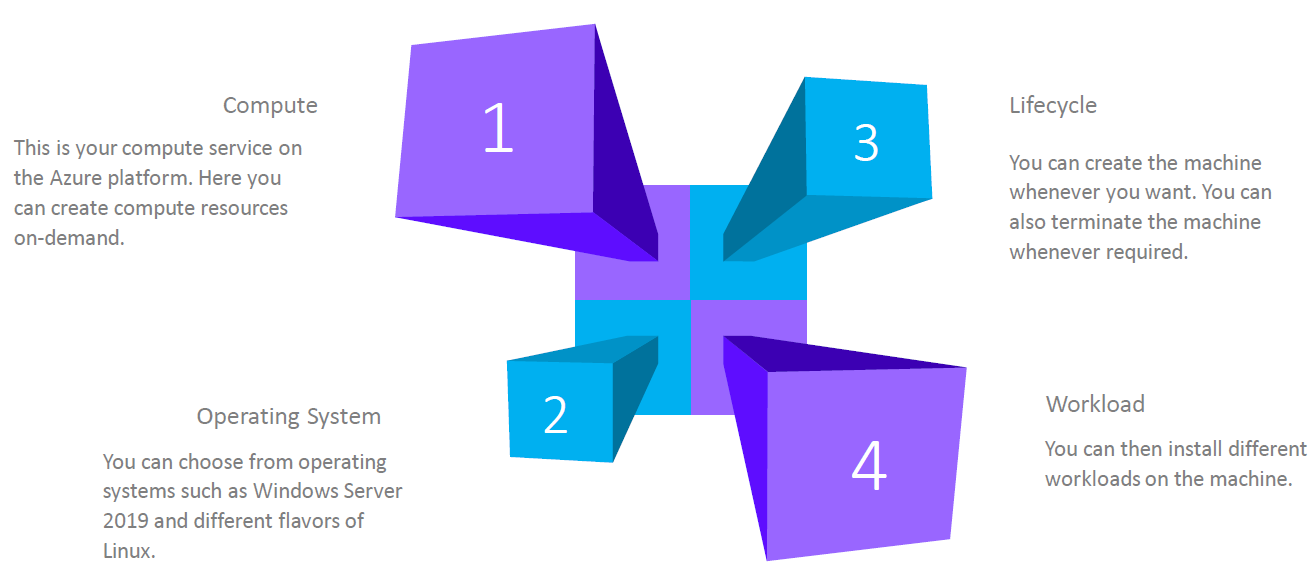
|  |  |
| --- | --- |
|  | * Now the question is If it all virtual – how it can we use it * Actually the VM can be created on a physical computer, and it will be called the host machine * VMs can also be created on servers or in the cloud * Azure VMs is a Iaas offring from Azure . The service let us launch VNs in Azure Cloud , hence giving us the fkexibility of virtualization witout having buy and maintain physical hardware. |

## WHY TO USE VIRTUAL MACHINES

|  |  |
| --- | --- |
| * Multiple VMs can be created on single physical server. |  |
| * We can create single purpose servers like testing , DB server etc without having to set up a whole physical computer. |  |
| * Create high level availability cluster and minimize downtime. * For that – lets say for high availibity of an application we can create multiple virtual machine and host the application on all the machine – So if any one of the VM is down – it won’ t be any downtime of the application. |  |

## AZURE VIRTUAL MACHINE SERVICE

* In corporate data centers, applications are deployed to physical servers, but we deploy applications in the cloud by renting(provisioning) virtual servers (Virtual Machine)



### AZURE VIRTUAL MACHINE TYPES

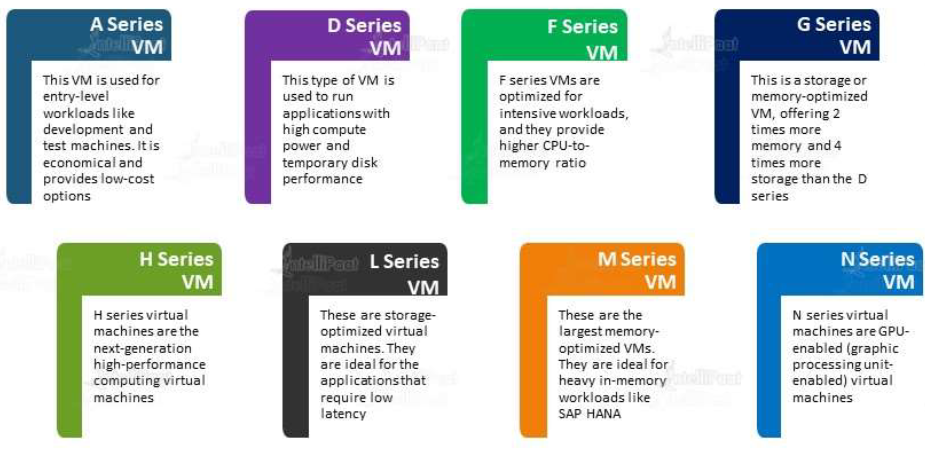
**The VMs in Azure are characterized based on the memory storage and compute types. Azure has assigned series for each such category**



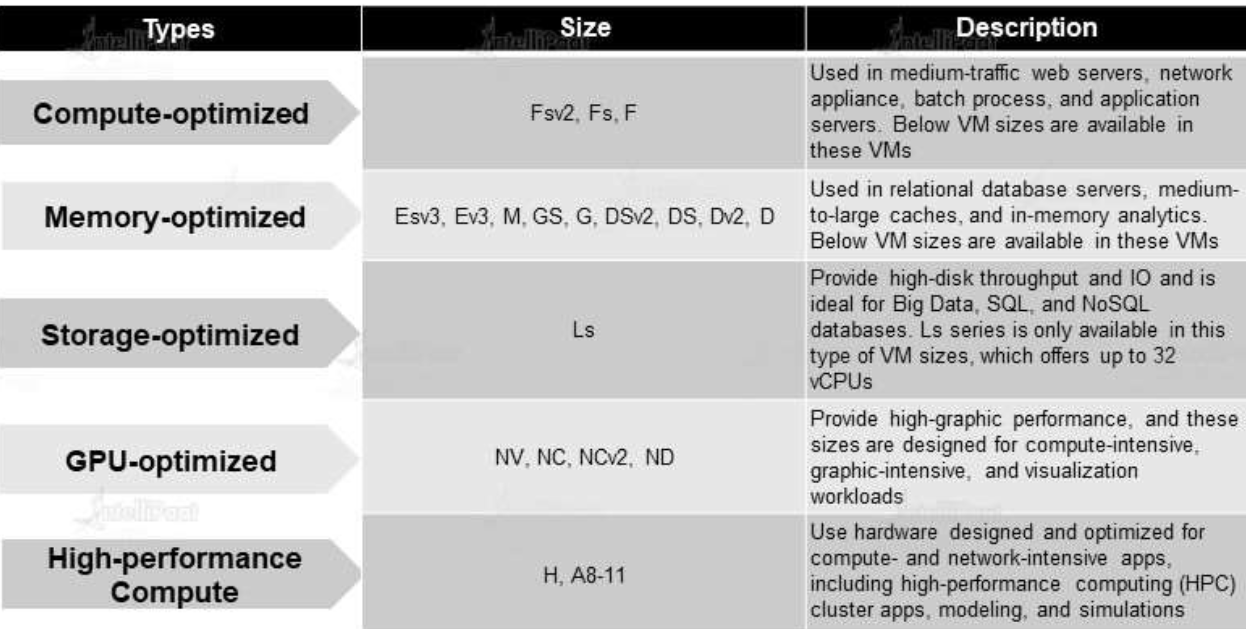
#### AZURE VM SERIES

* The Azure VMs are classified into different series based on the compute power, processing power , storage etc..
* <https://azure.microsoft.com/en-us/pricing/details/virtual-machines/series/>

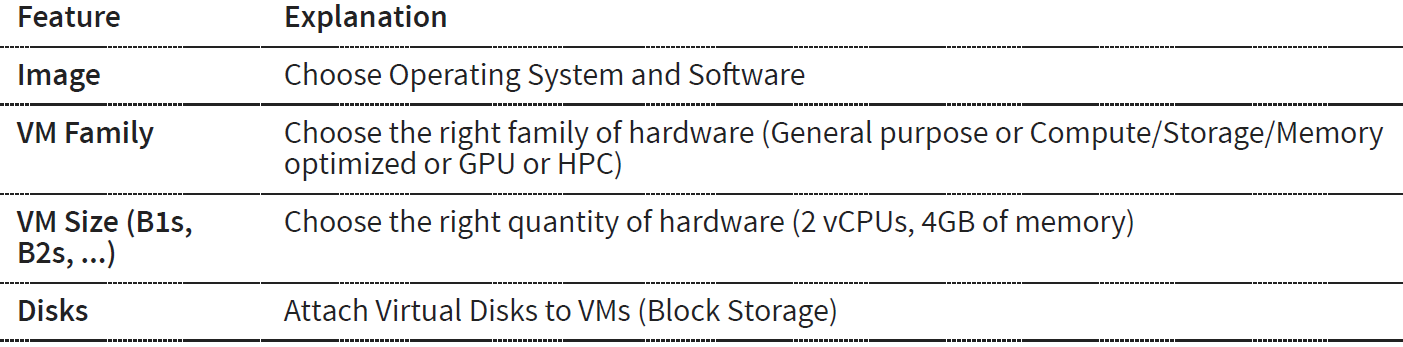
##### AZURE VM SERIES



***THE BELOW TABLE DESCRIBES WHICH VM FOR A SPECIFIC USE CASE***



### AZURE VIRTUAL MACHINE – KEY CONCEPTS



### CREATING/DEPLOYING A VIRTUAL MACHINE

#### DEPLOYING WINDOWS VM

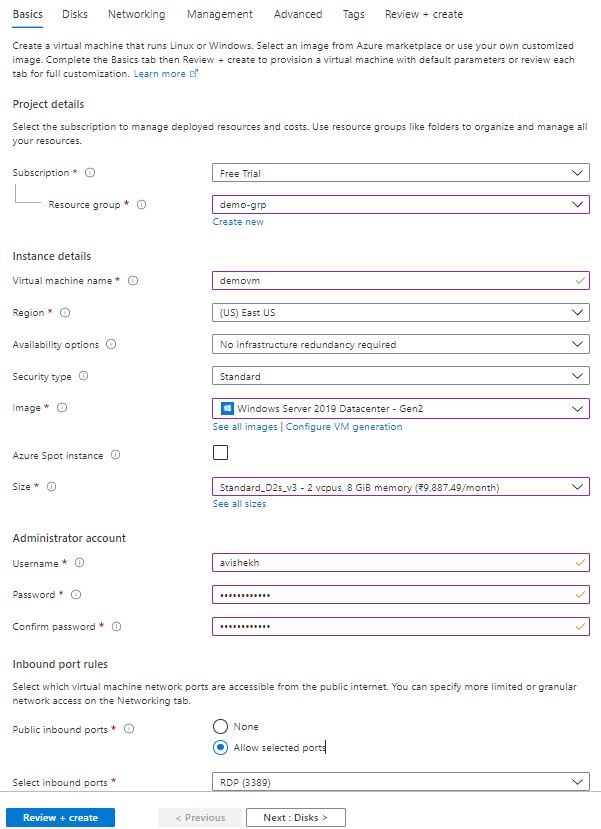
##### VIRTUAL MACHINE CONFIGURATION

* While creating a VM from azure portal we go throih



|  |  |
| --- | --- |
| RESOURCE GROUP | * The resource group of the VM * Even if the VM is created in resource group. We can move it to another resource group too. |
| VM NAME | Name of the Virtual machine |
| REGION | Region in which this VM will be created |
| IMAGE | * These are operating system image which can be used to install the OS in the VM * We can see the available images from “See All images”. This option will take us to marketplace where we have all the available images (OS Images) |
| SIZE | This defines the size of the resources in the VM like CPUs and RAM |
| SELECT INBOUND PORTS | * When we create a VM, by default all the connectivity to the VM is blocked. Hence, we need to allow the PORTs through which we can allow the connectivity * This configuration defines on which port the inbound traffic to VM is allowed.   For example – for windows we can use   * + For Remote Desktop – Enable port 3389   + For HTTP – Enable port 80 * For Linux we can use SSH |

##### BASIC TAB



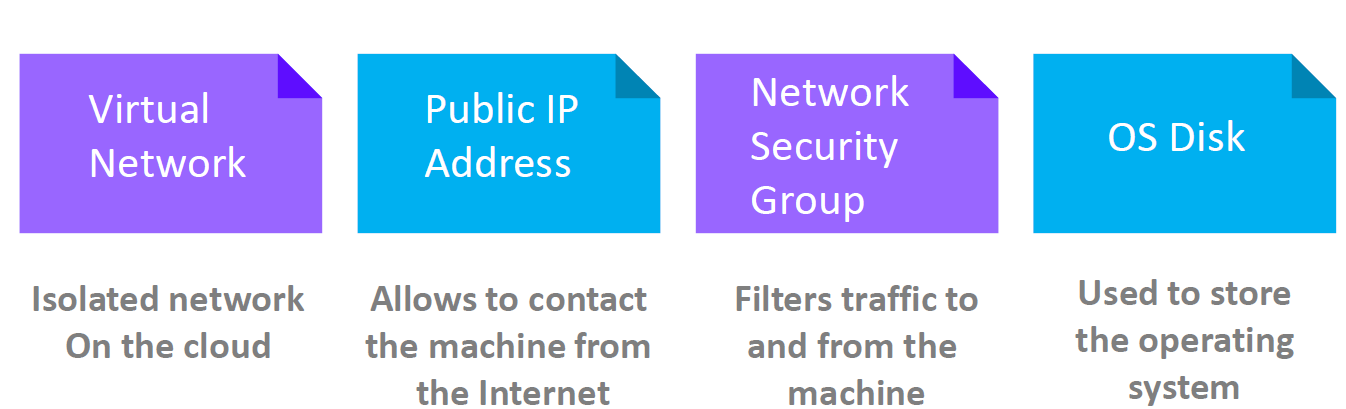
##### DISK TAB

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | |  |  | | --- | --- | | OS DISKTYPE | * This is the disk which will be used for the storage of data | | DATA DISK | * When we configure the VM – by default we get the OS disk. * For additional disk we can add as a Data disk to the VM | |

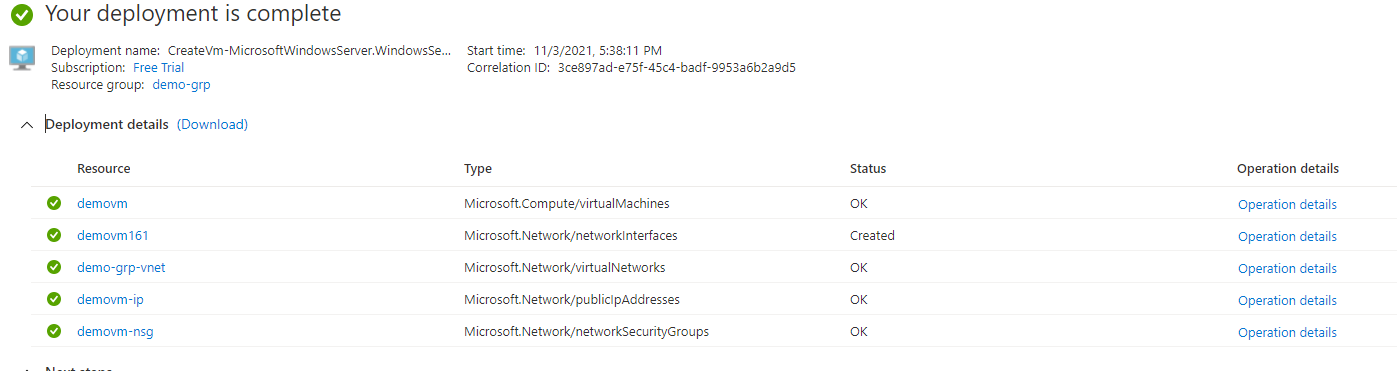
##### NETWORKING TAB

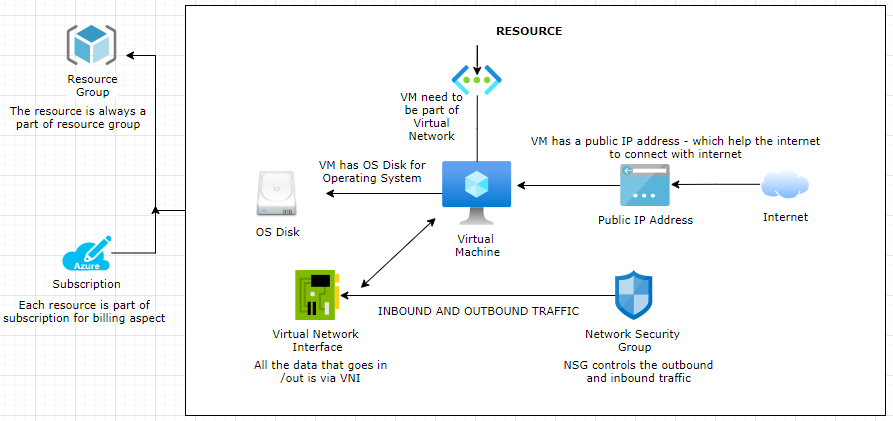
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  | | --- | --- | | VIRTUAL NETWORK | The VM needs to be part of a Virtual network. | | SUBNET |  | | PUBLIC IP ADDRESS | The VM can be accessed on Internet using its public IP address | |

##### RESOURCES CREATED WITH VM



* When a VM is created – then it creates multiple other resources as well. above are the resources created with VM
* The below diagram shows the created VM and other related resources which are created with the VM like NIC, VM, NSG and public IP address.





|  |  |
| --- | --- |
| OS DISK | * The virtual machine has OS Disk where operating system can be installed * Addition disk can be added too |
| VIRTUAL NETWORK INTERFACE | * Virtual Network Interface is like network interface card * All the data that goes in or out go via Virtual Network Interface |
| VIRTUAL NETWORK | * Every VM is part of a Virtual Network. |
| NETWORK SECURITY GROUP | * It controls all the inbound and outbound traffic to and from the VM |
| PUBLIC IP | * The VM are always associated to a public IP address –through which the internet can connect with the VM |

* To delete a VM: From the *menu 🡪 All Resources*. Select all the resources which are related to the VM 🡪 Type yes and Delete

##### CONNECTING TO VM (WINDOWS) USING RDP

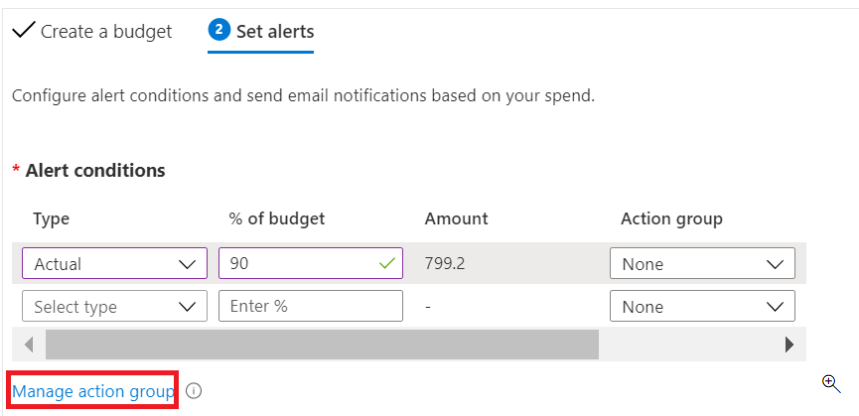
|  |  |
| --- | --- |
|  | * To connect with Windows based VM we use RDP. * The Public IP of the VM is used to connect to the VM * SSH is used to connect to Linux based VM |
|  | * To connect to the windows VM- download the RDP file * Open the downloaded RDP and Enter the username/password (used while creating the VM)   Alternately we can use windows “***mstsc***” service to connect with the VM |

##### COST ASSOCIATED WITH A VM

* When a VM is created there are multiple resource gets created along with it. The resources like *Virtual Network, Network Interface card and Network security Group* has no cost associated with them
* But for other resources which as cost associated with it e.g., *VM, IP and the disk* – we can set up an email notification when a threshold is reached.
* Go to Subscription 🡺 Budget 🡪Add

#### INSTALLING APPLICATION IN VM(IIS)

To install an application, we have multiple options. For example, to install IIS, the options can be

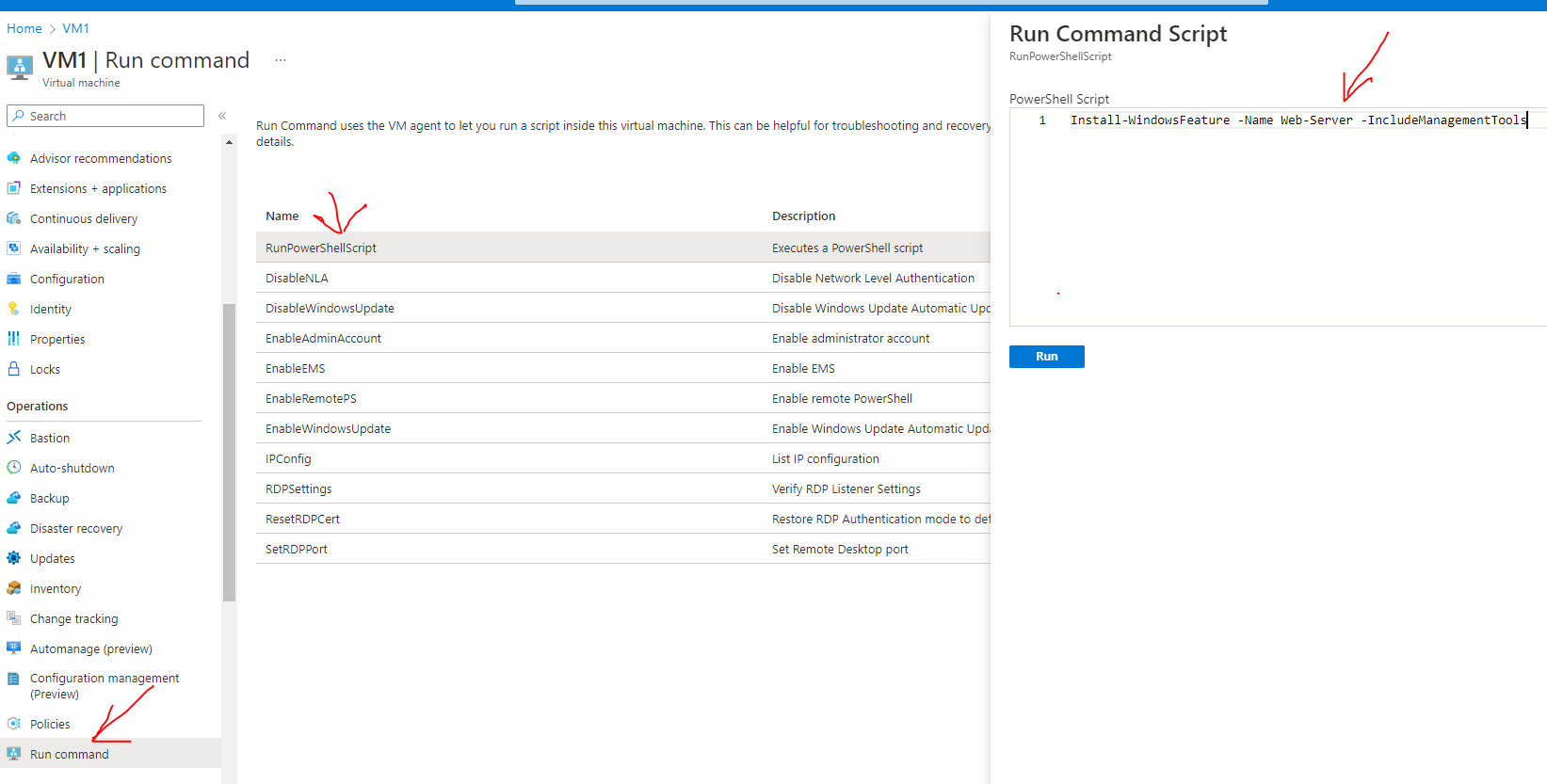


1. *SERVER MANAGER DASHBOARD*
2. *FROM THE VM ITSELF USING POWERSHELL*
3. *RUNNING SCRIPT FROM AZURE PORTAL*

*Note – For IIS to access we need to enable HTTP ports and to log in we need RDP access*

##### FROM AZURE PORTALS

* Go to the *VM 🡪 Run Command 🡪 RunPowerShellScript 🡪 Enter the windows Command 🡪 Run*
* Command : **Install-WindowsFeature -Name Web-Server -IncludeManagementTools** *(Note – This is not an Azure command. It’s a windows command)*
* This will Install IIS which can be accessed using via Public IP of the VM



#### DEPLOYING LINUX VM

##### HOW SSH WORKS

* SSH stands for Secure Shell
* It’s a protocol to allow 2 or more machine to communicate over a network in a secure way.
* The secured communication between machine is possible because the data exchange happens through encryption.
* For the communication to happen the server should have SSH enabled (e.g., windows /Linux should have openssh installed on the server). The client must have a SSH client e.g., Putty for windows or terminal for Mac/Linux.
* For the communication we need to authenticate the connection. Authentication can be done in 2 ways
  + Password based authentication
    - For password-based authentication we use
      * *user@IP\_of\_the\_remote\_server*
      * *Then provide the password*
  + Public and private keys-based authentication

**PUBLIC AND PRIVATE KEY BASED AUTHENTICATION**

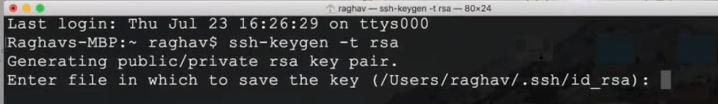
*Step 1*: Generate the public and private keys

Let’s understand it with an analogy. Let’s say 2 people want to send a box

|  |  |
| --- | --- |
| 1. The sender will put a lock on the box and send it to receiver | 2.When receiver gets the box, Since the sender has key to the lock hence receiver cannot open it. |
| 3 – Now the receiver – He put his own lock and send back to sender. | 4- Once the sender recieves the box. He opens the lock using his own key and send back to reciever. |
| 5.Once the receiver gets the box. Now he can be able to open it with his key | **In the entire communication – the box has always has lock on it – which makes it secure.** |

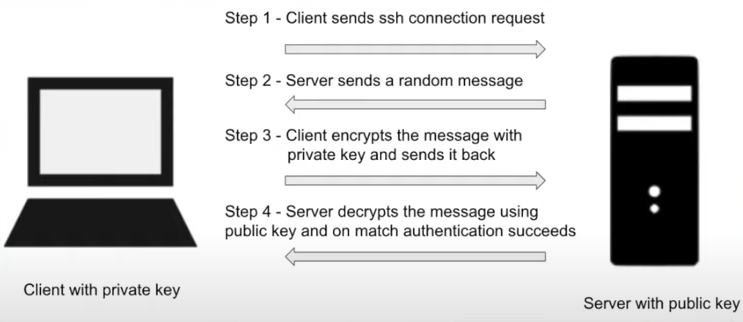
HOW SSH FITS IN THE ABOVE EXAMPLE

* Step 1: Generate a Public and private key using – ***ssh-keygen -t <encryption algorithm>*** (e.g., rsa/MD5)



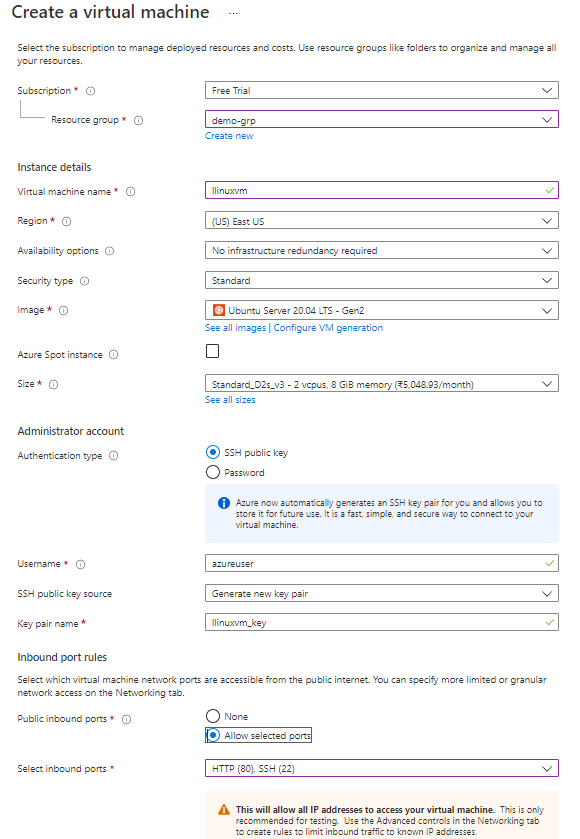
|  |  |
| --- | --- |
| * Step 2: Copy the public key to the remote SSH server. Hence the final set up will look like as shown below |  |

* Step 3: For authentication below are the steps client / server follow for authentication using public and private key.



##### DEPLOYING LINUX VM (USING SSH KEY)

* For the public -private key pair authentication – we store the public key in the VM itself
* The client machine which will connect to it - will have the private key – which will be used for authentication into the Linux VM.



##### CONNECTING TO *LINUX* VM *USING* SSH

* The authentication type for a Linux VM can be
  + PASSWORD BASED AUTHENTICATION
  + SSH KEY BASED AUTHENTICATION

###### SSH KEY BASED AUTHENTICATION

The public -private key pairs can be generated two different ways

1. USING CLOUD SHELL (<https://learn.microsoft.com/en-us/azure/virtual-machines/linux/mac-create-ssh-keys> )
2. PUTTY GEN TOOL

CLOUD SHELL

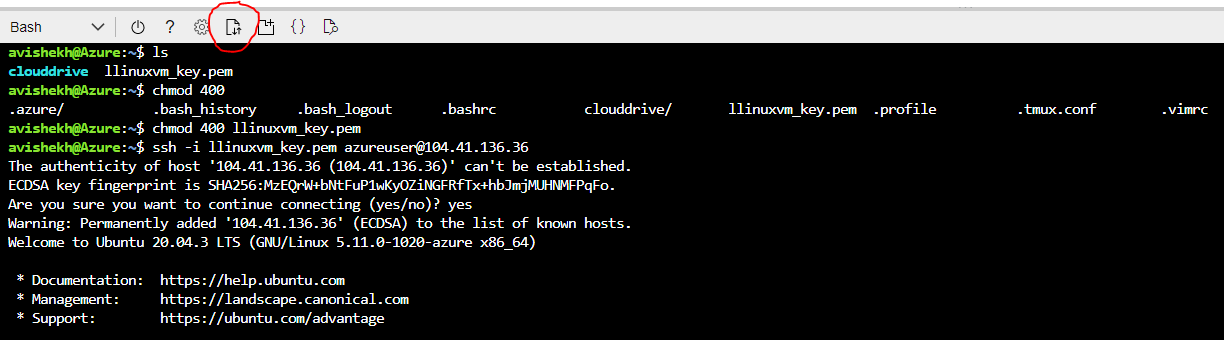
|  |  |
| --- | --- |
| Step 1: Create a Linux VM 🡪 In “SSH public key” we need to paste the public key |  |
| Step 2: Generate the public key from the CLOUD Shell  Command: ***ssh-keygen -t rsa -b 2048*** |  |
| Step 3: Navigate to public key |  |
| Step 4: Copy the Public key content and paste in the SSH public key field  Step 5: Review & Create -🡪 Create |  |
| Step 6 – Login to the VM  To verify whether we have logged in to correct VM is the “hostname” command | Command: user@publicIpOfVM  Example: ***avishekhssh1@20.228.92.241*** |
| We can install software as well like Apache. It can be accessed using public Ip of the VM | **COMMAND**  *sudo apt-get -y update*  *sudo apt-get -y install apache2* |

PUTTY GEN TOOL

* We can generate the public-private key pair using using putty gen too as well and follow the same process as above

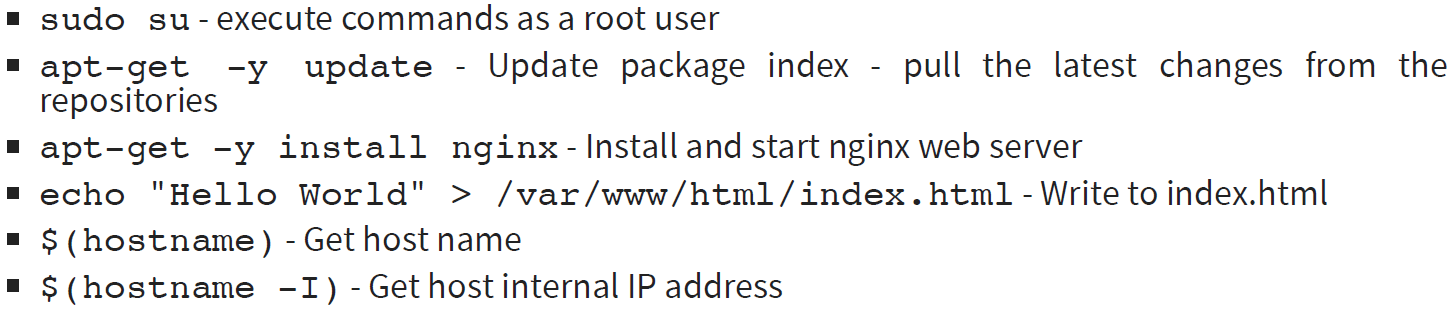
To login into Linux VM we need to use Putty Tool.

|  |  |
| --- | --- |
| * Open the Cloud Shell and choose Bash * Select the subscription 🡪 Create Storage |  |

* Upload the SSH key using upload option 
* Navigate to VM 🡪 Connect 🡪SSH. Follow the steps to SSH the VM

#### INSTALLING SOFTWARES IN VM

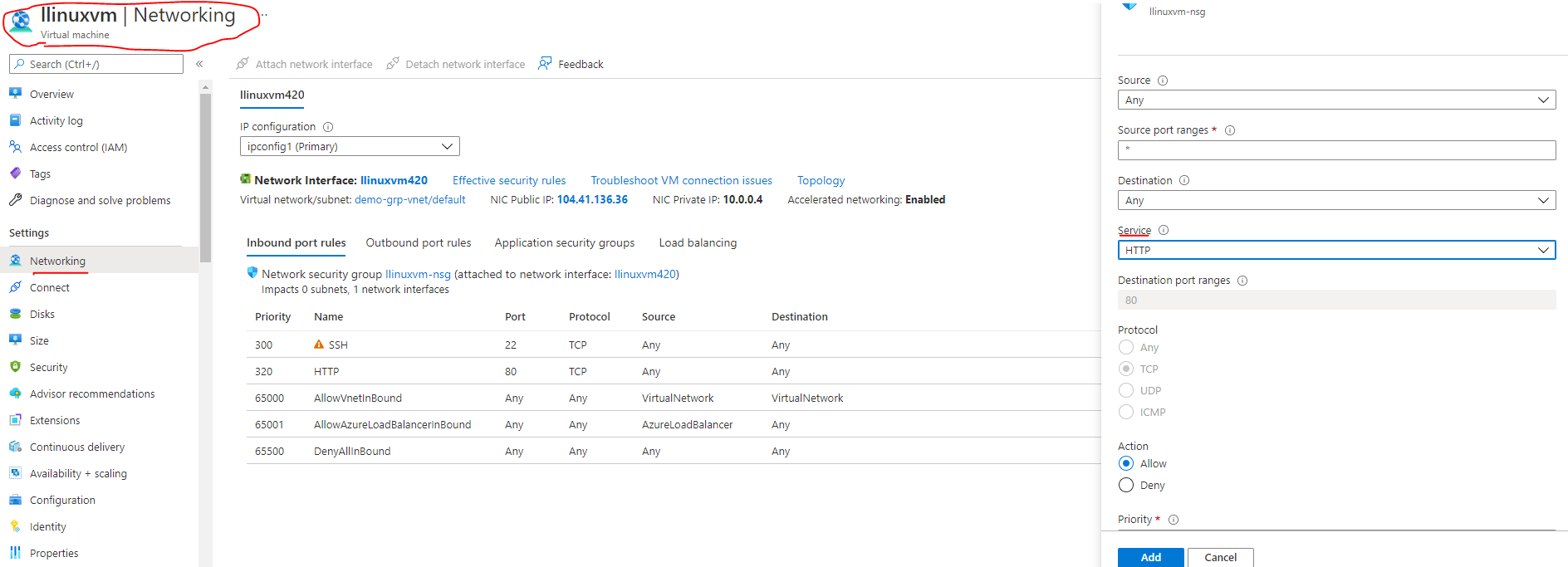
##### INSTALLING NGINX (HTTP SERVER)



* The nginx webserver can be accessed using the public IP address of the VM (<http://104.41.136.36/> )

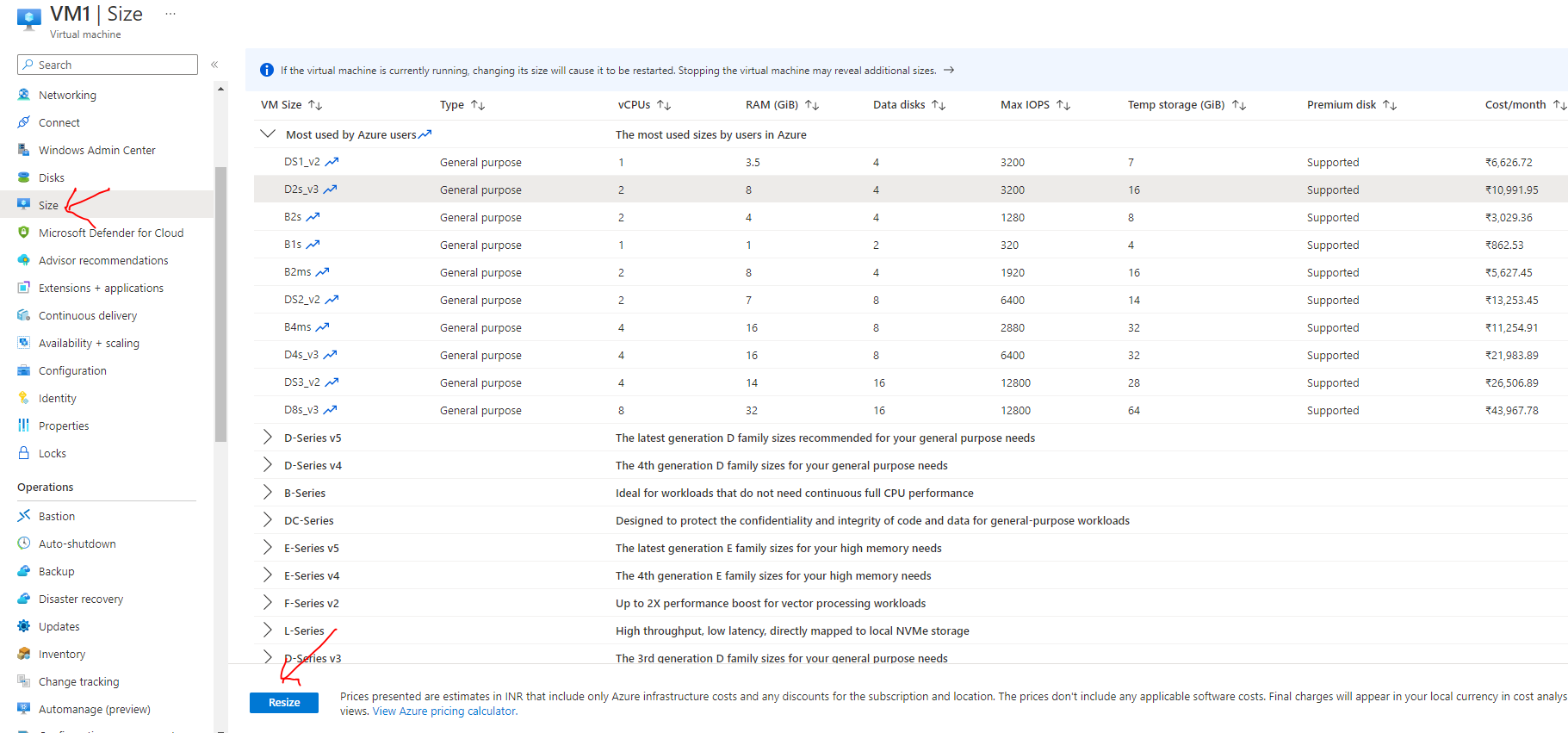
###### ADDING INBOUND PORT RULE

* We can add the inbound port rule to accept the incoming request. For example – we can enable the HTTP service at port 80 by adding a new inbound port rule.
* The same can be done while creating the VM (***Basic 🡪 Select inbound port***)

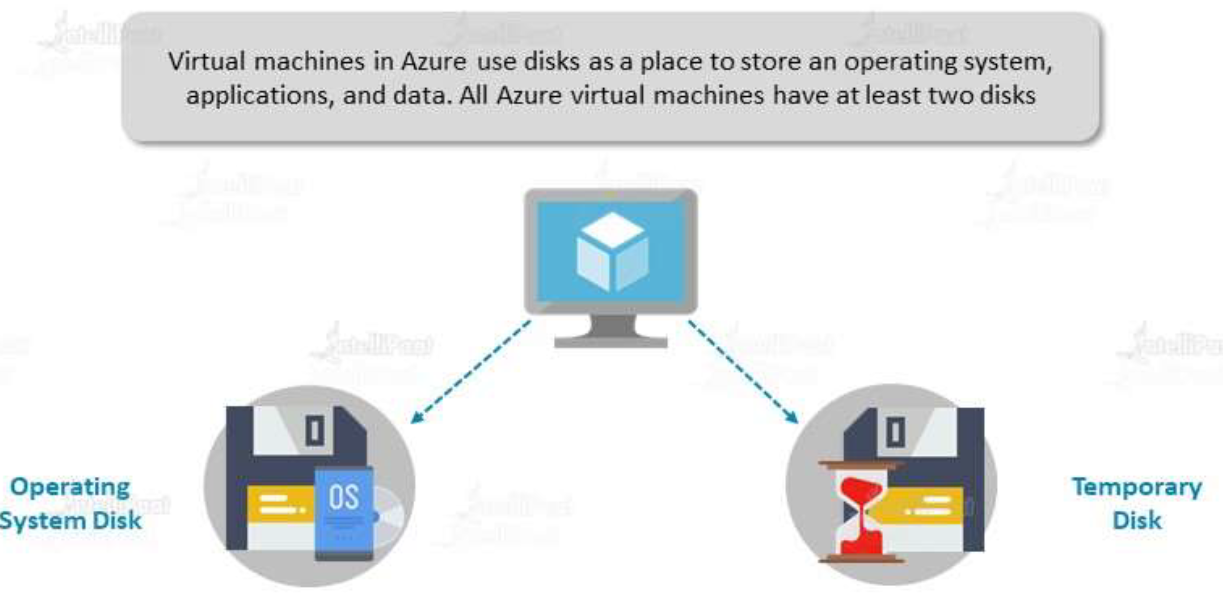


###### RESIZING THE VM

* Go to VM Resource 🡪 Size 🡪 Select the Size 🡪 Resize



### VIRTUAL MACHINE DISK



|  |  |
| --- | --- |
| When we deploy a VM is Azure, this get associated with Disk. The disks in Azure can be classified as   1. *OS DISK* 2. *TEMPORARY DISK* 3. *DATA DISK* – This is a separate resource in Azure, which can be attached to the VM.   These disks are further classified as   1. *MANAGED DISK* 2. *UNMANAGED DISK* | FEATURE OF VM DISK |

#### OS DISK

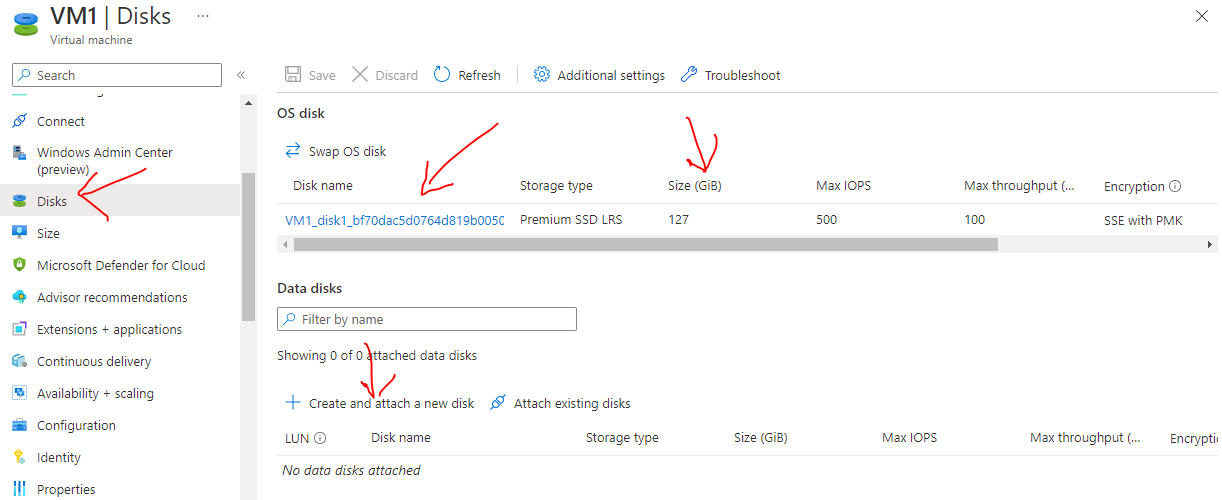
* When we create a VM – along with an OS Disk, it gets a temporary storage as well. **The OS disk size called managed disk as this is completely managed by Azure itself.**
* The OS disk are created from a VHD (Virtual Hard Disk) file – which is an image of the Operating System
* The managed disks are designed for availability – where the data is replicated to multiple different location within the data center for high availability
* The size of temporary storage depends on the type of VM we are using. Some VM has temporary disk associated – some might don’t have.

##### DISK MENU

* From disk menu option - We can see the details of disk attached to the VM and attach a new data disk to VM

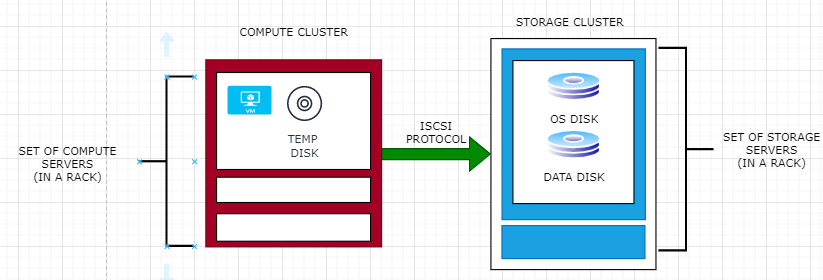
DEFAULT OS DISK SIZE

|  |  |
| --- | --- |
| **VM TYPE** | **DEFAULT OS DISK SIZE** |
| Windows VM | 127 GB |
| Linux | 30 GB |
| SMALL DISK(WIDDOWS) | 30 GB |



#### TEMPORARY DISK

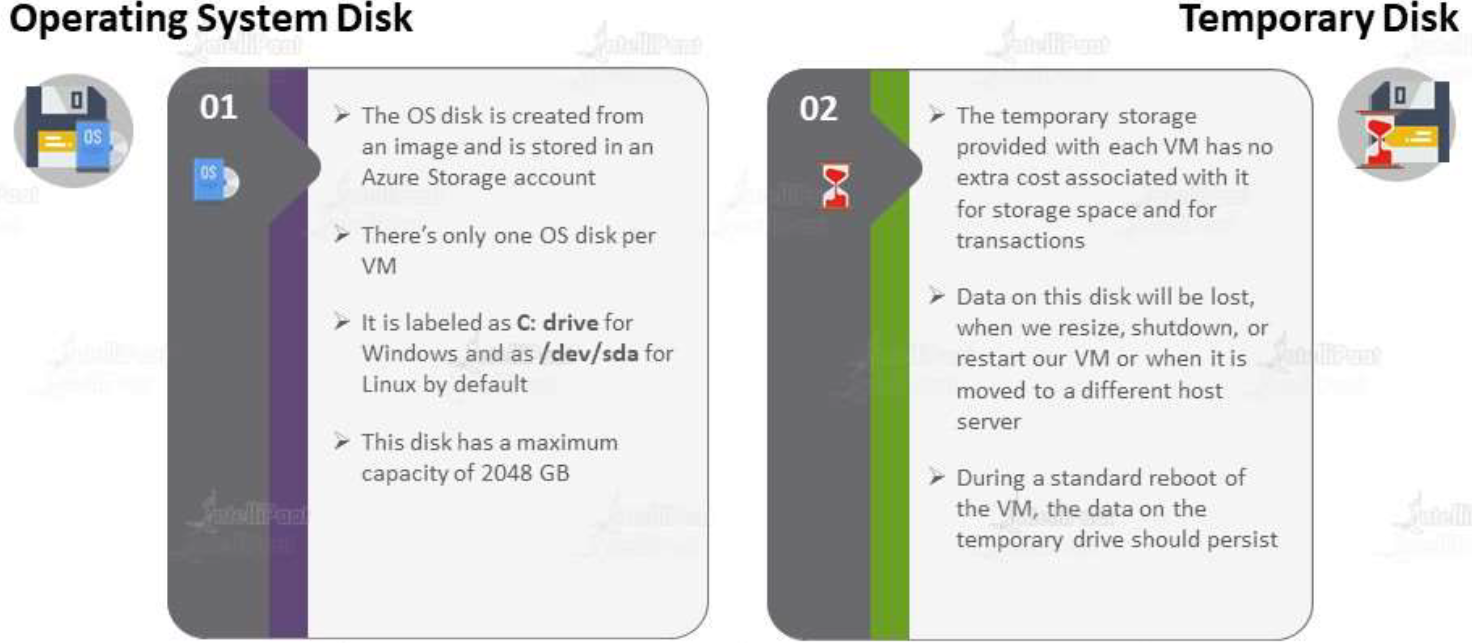
* When a VM is created – Azure will allocate a capacity on the physical hardware.
* The data and OS disk strored in the physical storage e.g SAN.
* Hence there is seperation between – where the compute utilization and stoarage capacity is coming from.
* The communication between then is done using ISCSI Protocol.
* The teampory is part fo the allocated space of compute capacity (VM allocated space) itself not from the storage cluster. Since temp disk is part of compute capacity , data can be accesses with low latency.



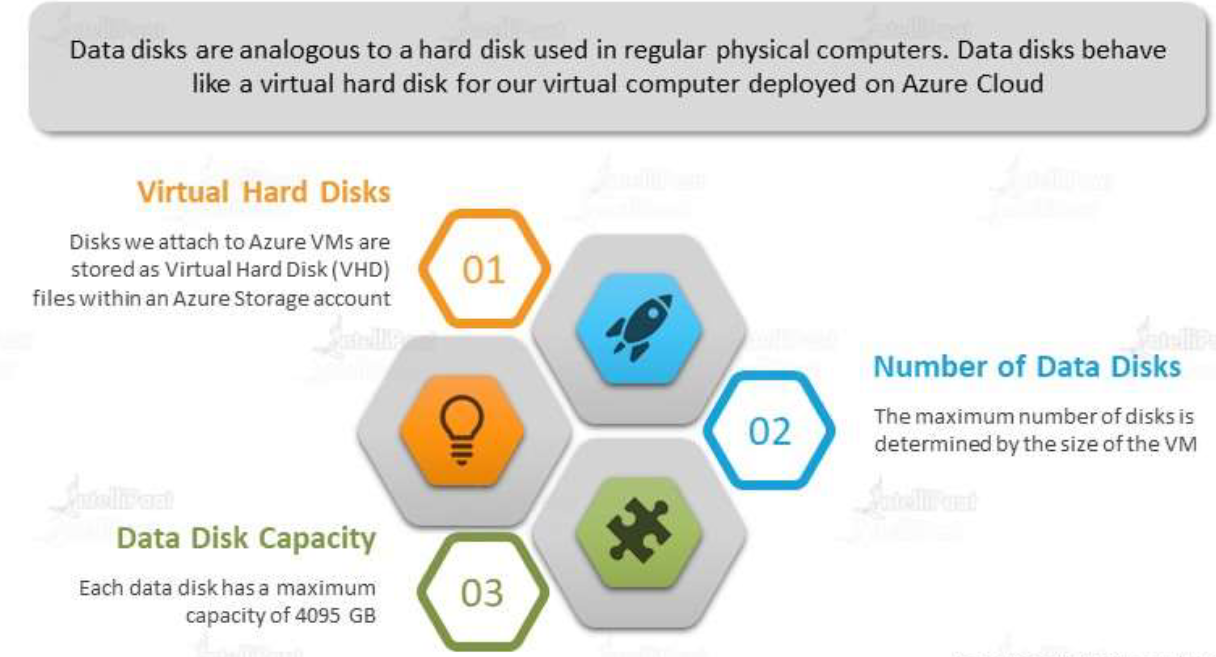
* Now, when the VM is stopped – a deallocation will happen for the VM. Azure will deallocate the VM from the current physical hardware
* On Restart – The VM will be allocated with a new location in a different physical hardware. In the entire deallocation process – the OS and Data disk will not be impacted as they not part of the compute.
* **Since the VM is allocated with new location on physical hardware– all the data in the temporary disk will be lost**

*Note – When the VM is deallocated – there will be no compute cost for the VM. But only charged for storage capacity.*

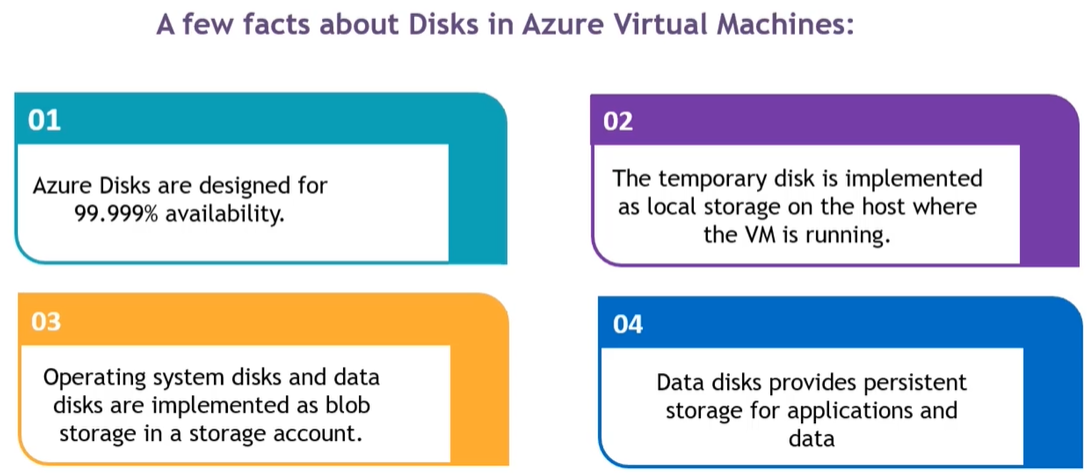
#### DIFFERENCE BETWEEN A OS AND TEMPORARY DISK



#### DATA DISK

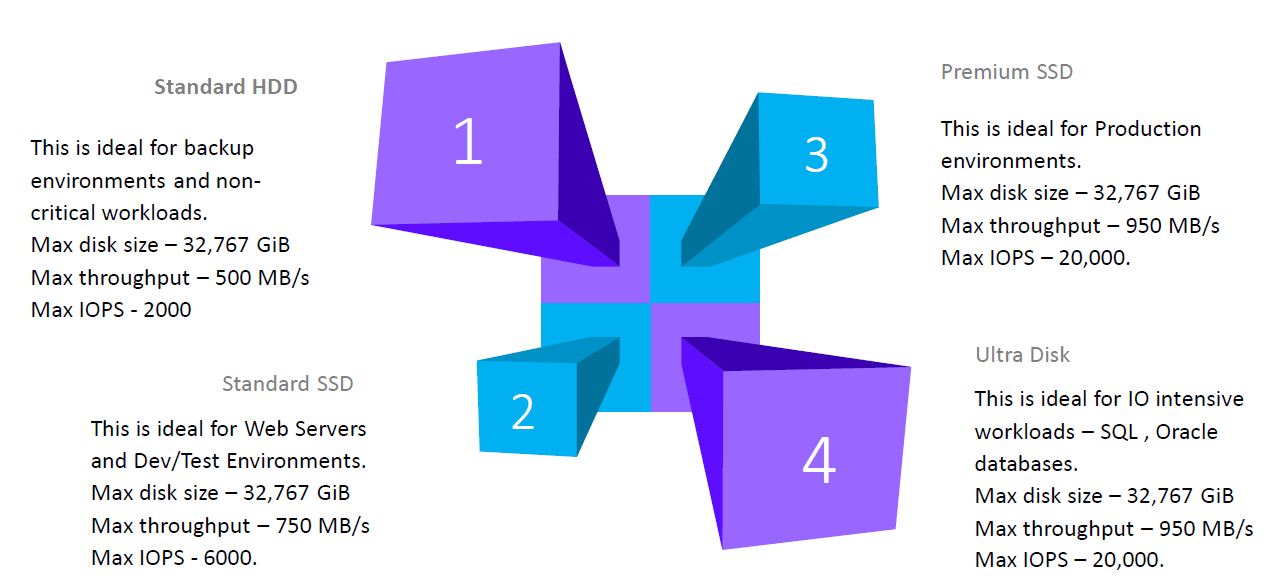


* Azure data disk are managed disks. these disks are like physical disks but can eb virtualized
* Designed for high availability (99.999%) and has support for features Availability Zones, Azure Backup etc..



##### TYPES OF DISKS (BASED ON PERFORMANCE)

***The disks in azure can be classfied based on their performance.***



|  |  |
| --- | --- |
| *\*****HDD*** *-Hard Disk Drive*  *\*****SSD*** *– Solid Disk Drive* |  |

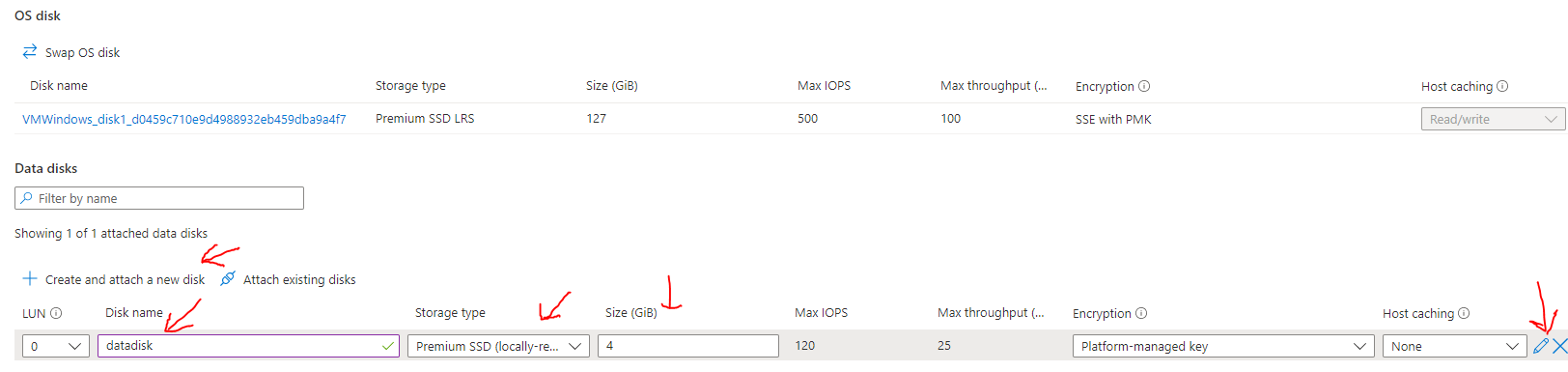
##### DATA DISK LAB

* The maximum capacity of the data disk that can be attached to a VM is 32 TB for a single disk.
* To increase a capacity of the existing disk – the VM should be in stopped (deallocated) state. Although new data disk can be attached to a VM , even it is in running state.
* Reducing the size of data disk is not possible – as it might lead to data loss.

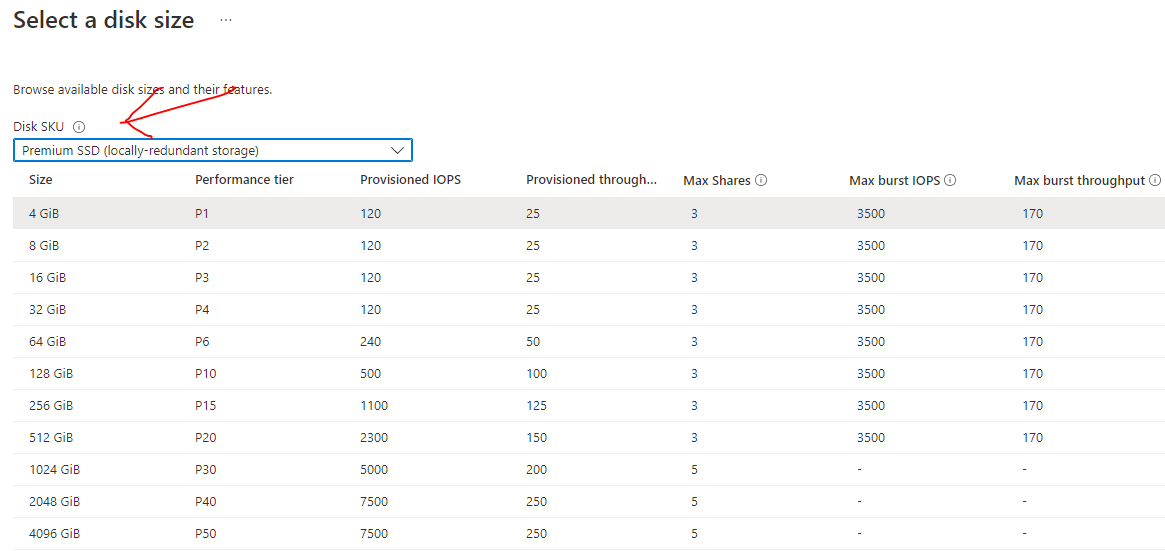
Creating a Data disk – it’s a 2-step process

###### CREATE A NEW DATA DISK

* Go to Disk Tab 🡪 Create and attach a new disk
* LUN (Logical Unit Number) - Logical unit number of the data disk. This value is used to identify data disks within the VM and therefore must be unique for each data disk attached to a VM.



|  |  |
| --- | --- |
|  | * Data disk can also be added to the VM even after he VM is created using Disk option in Menu * This will be a separate resource in Azure * The reason, it’s a separate resource is that it can be detached from VM at any point of time. The disk can be then assigned to another VM * Select the disk size + the Disk SKU 🡪 *Save* |



###### ATTACHING(MOUNTING) A DISK TO VM

* Login to VM 🡪 Run 🡪 ***mstsc*** 🡪(RDP)🡪 Server Manager 🡪 File and Storage Services 🡪 Disks Menu 🡪 Right Click on recently attached disk 🡪 New Volume

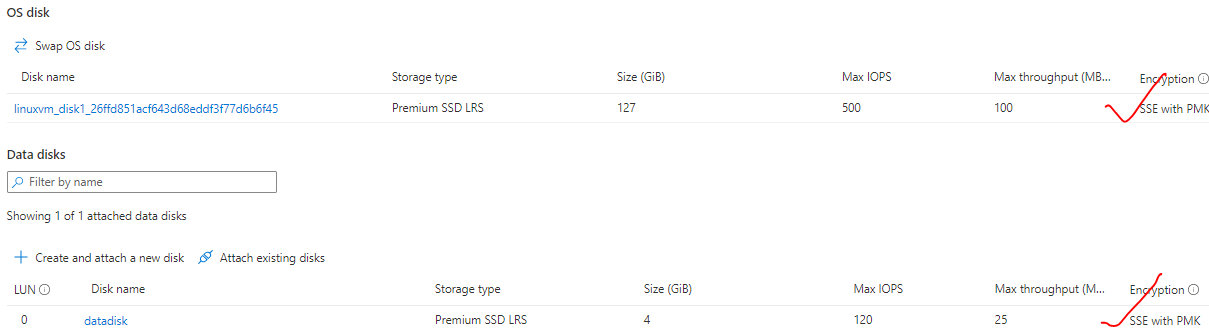
|  |  |
| --- | --- |
|  |  |
|  | |
|  | |
|  | |
|  | |

#### DATA DISK ENCRYPTION

* <https://learn.microsoft.com/en-us/azure/virtual-machines/disk-encryption-overview>

##### SERVER-SIDE ENCRYPTION

* The organization has requirement – that the data stored in the Azure data center needs to be encrypted in rest i.e. This means that when the data is finally stored on the disk in Azure data center on these physical disks, it needs to be encrypted.
* For this Azure has a security feature called *Server-Side Encryption*
* This feature is available for both data and OS Disk as shown below. **The disks are Server Side Encrypted with PMK (Platform Managed Key)**



* When we want to encrypt data, we need to use an encryption key along with an algorithm for encrypting the data.

For Azure managed encryption the keys are being managed by the Azure platform themselves.

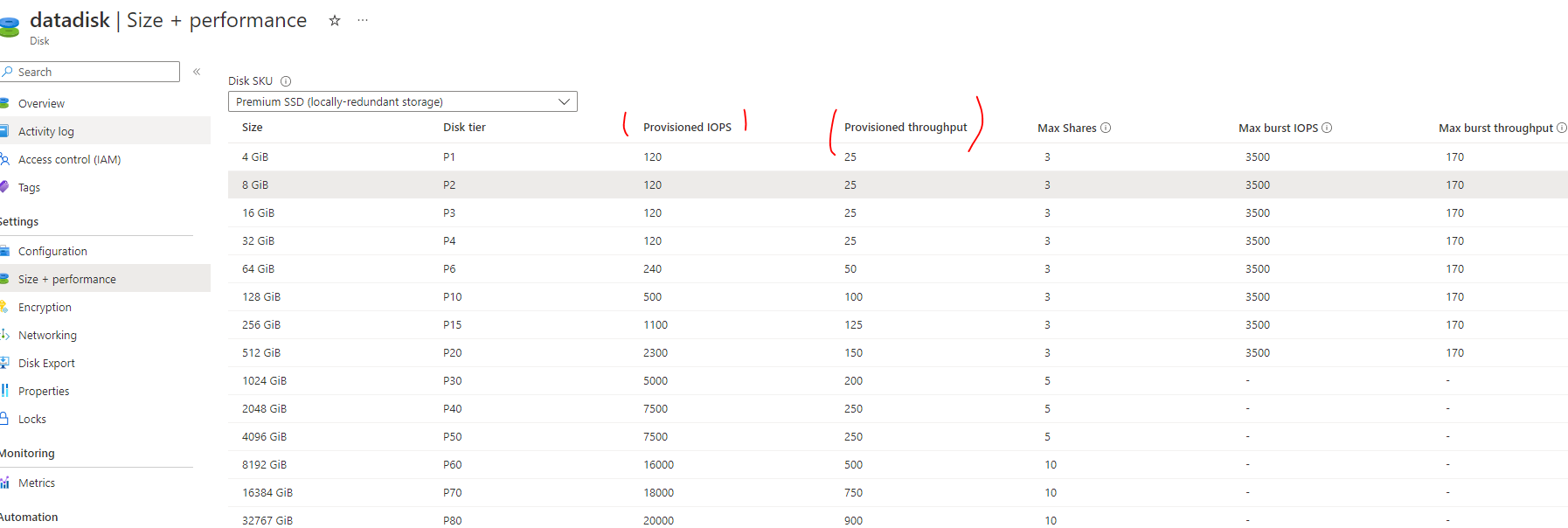
* Azure also gives the ability to use customer managed keys where organizations manage the encryption keys.

###### SERVER SIDE ENCRYPTION USING CUSTOMER MANAGED KEY

* In the above example – the key is getting managed by Azure Platform itself. But the the encryption key managed by the customer also.
* To store the customer encryption keys – we can make use managed service known as key vault, as your keyboard is a mode service in which you can store your encryption keys

##### AZURE DISK ENCRYPTION

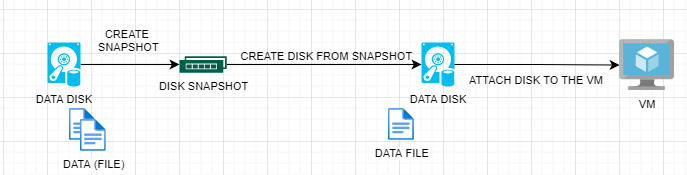
#### IOPS AND THROUGHPUT



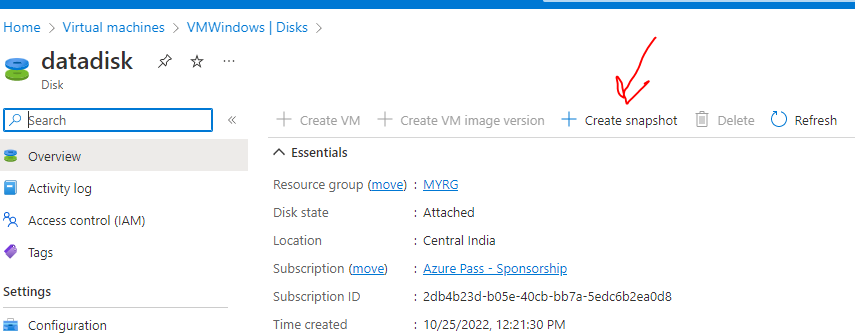
* **IOPS** – Input /Output Operations per second. Example IOPS of 120 means – we can do 120 input/output operation on a given disk.
* **THROUGHPUT** - Throughput is a measure of how many units of information a system can process in a given amount of time.

#### DATA DISK SNAPSHOT

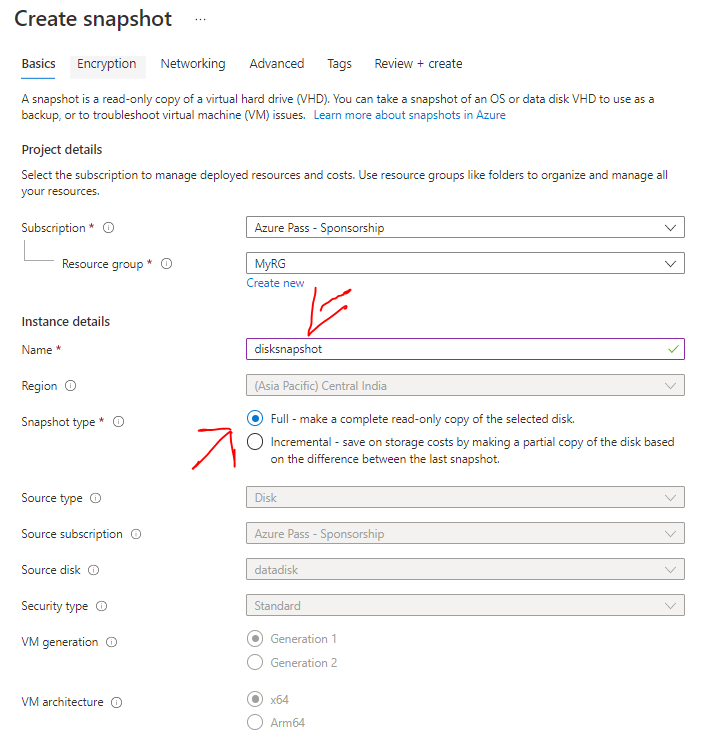
* We can take a snapshot of an existing disk and then we can create a new disk out of that snapshot.
* For example – Let’s say we have a data disk attached to a VM. If we have some data already there on the disk. Using the disk snapshot feature, we can take the snapshot of the disk
* If create a disk using the snapshot – it will have the data – which was there when the snapshot was taken.
* The disk snapshot can be attached to multiple VMs – but it cannot be considered has shared disk because – it will have the data – which was there when the snapshot was taken.



##### CREATING A DISK SNAPSHOT

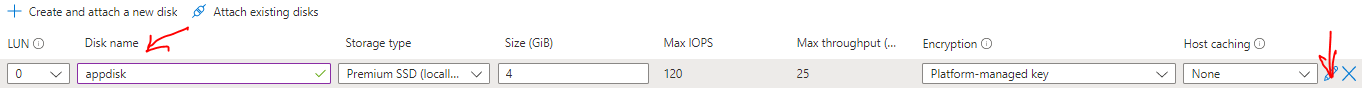


* Give a name of disk snapshot, Snapshot type and Storage type (e.g. . Premium SSD)
* Review & Create 🡪 Create



##### ATTACHING A DISK VM CREATED FROM DISK SNAPSHOT

We can create a disk out of this Snapshot – and then attach it to a VM



|  |  |
| --- | --- |
|  | * Storage Type – Snapshot * Snapshot 🡪 Select the previously created snapshot * OK 🡪 Save   Now when we login to VM - to which the newly created disk has been attached.  We can see the data available on newly created disk - which was there while creating a snapshot. |

#### AZURE SHARED DISK

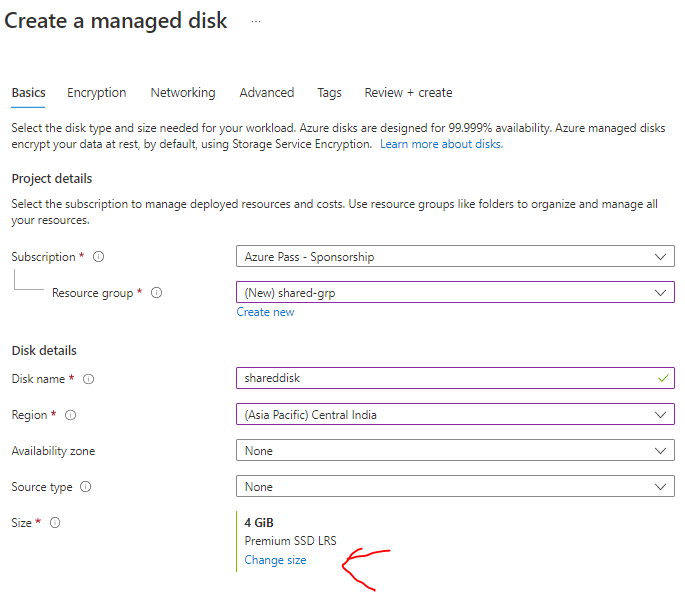
|  |  |
| --- | --- |
|  | * This allows a managed disk to be attached to multiple VMs   *Restiction* –   * ***This can be enabled for Premium and Ultra Disks*** * If VM-1 has has shared disk and we want to attach the shared disk to VM-2 . Then VM-1 needs to be stopped before attaching the shared disk   Note – If the disk is shared – it does not mean that the file creaed from VM in the disk will be visible in another VM. We need some sort of software to sync the data / files |

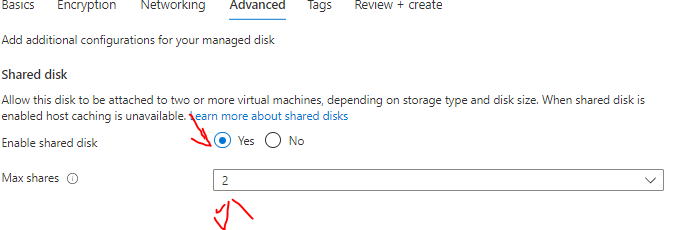
USE CASE –

* By default - when we create a disk, it can only be used by one VM at a time. But when we have a workload such as a clustered SQL server workload, we might have to install Microsoft SQL Server Database engine on multiple VMs to ensure that we have more compute power. These SQL server nodes (VM) need to access a single data source, so the data for that database engine reside only on one disk.
* In this case we data source can be a Shared Disk

##### CREATING A SHARED DISK

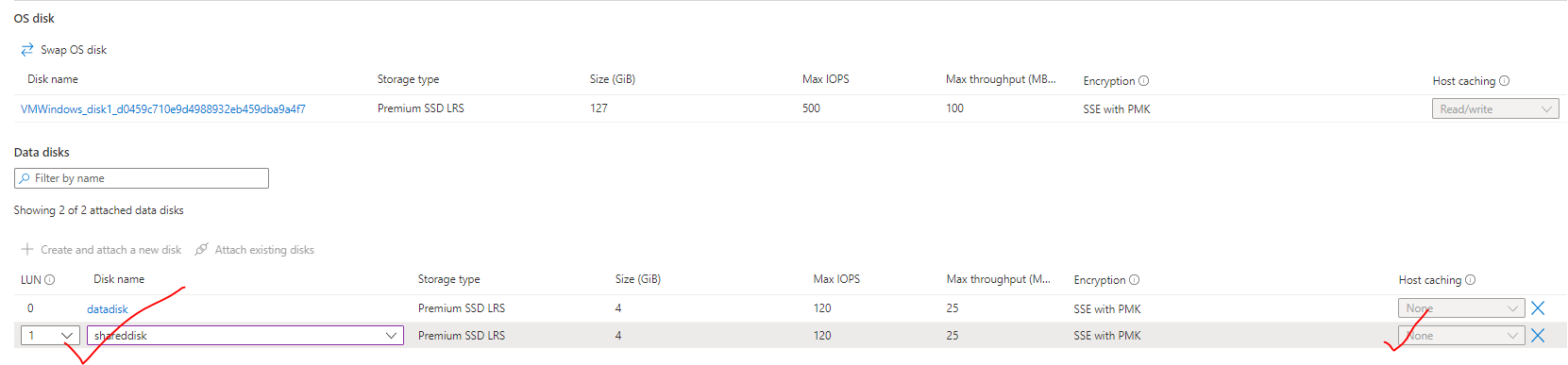
* Create a resource 🡪 Search for “Managed Disk” 🡪 Create





##### ATTACHING TO VM

* If we have an existing VM , then the VM needs to be stopped before attaching the shared disk
* Note – First select the “Host Caching” then select the “Disk name”



#### CUSTOM SCRIPT EXTENSIONS

* This tool can be used on Azure virtual machines to download and execute scripts.
* ***This is ideal when we want to deploy any custom configuration of any software installation on a virtual machine.***
* The scripts can be in an *Azure storage account or even in GitHub*.
* A time duration of 90 minutes is allowed for the script to run. Any longer and the result will be a failed extension provision.
* It’s ideal not to place reboots inside the script, because the extension will not continue after the reboot. Hence if you have other commands that need to run via the extension after the reboot, they won’t run.
* If your script does need a reboot, then maybe you can look at other tools such as Desired State Configuration, Chef or Puppet.
* The script will run only once.
* The Custom Script Extension will run under the impersonation of the LocalSystemAccount.

##### CREATE A STORAGE ACCOUNT

* As the custom script can be stored in storage account or GitHub

|  |  |
| --- | --- |
| **Create a Storage Account 🡪 Create a Container 🡪 Upload the Powershell Script** | ***POWER SHELL SCRIPT (TO INSTALL IIS)***  *import-module servermanager*  *add-windowsfeature web-server -includeallsubfeature*  *add-windowsfeature Web-Asp-Net45*  *add-windowsfeature NET-Framework-Features* |

##### CREATE A LINUX VM WHERE SOFTWARE WILL BE INSTALLED

|  |  |
| --- | --- |
|  | ***SHELL SCRIPT (TO INSTALL NGINX IN LINUX VM)***  apt-get update -y && apt-get upgrade -y  apt-get install -y nginx |
| * For Linux VM we need to ***install Custom Script for Linux*** * Browse to the Shell Script in the Storage account container * Enter the command “sh <***file\_name***>” * The ngnix webserver can be accessed using public IP |

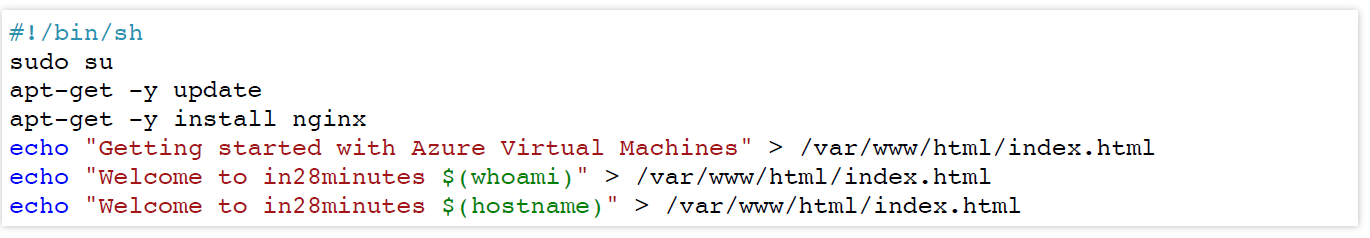
##### CREATE A WINDOWS VM WHERE SOFTWARE WILL BE INSTALLED

|  |  |
| --- | --- |
| * *Step 1* - Create a VM * *Step 2* - Advanced Tab 🡪Select an Extension to install🡪 Search For “Custom Script Extension” 🡪 Install * *Step 3* - Upload the Script File from the container of the storage account * *Step 4* -Review & Create 🡪 Create * *Step 5* – IIS will installed to the VM with its creation. * *Step 6* – The IIS can be accessed via VM’s public IP |  |

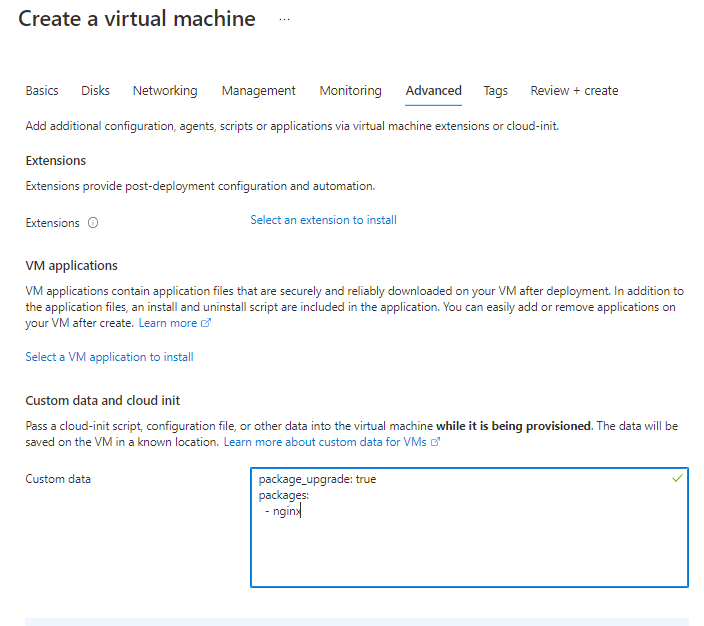
|  |  |
| --- | --- |
|  | * Browse to the Shell Script in the Storage account container     *CUSTOM SCRIPT UPLOADED* |

#### USING CLOUD INIT

* In the above steps – We first created the VM and the installed the nginx server. We did all by doing the “***ssh***” and running command from cloud shell.
* Just In case if we want to run a specific set of command after the VM start up – we can write the series of command in “Advanced Tab” 🡪 Cloud init as a bash script. As show below.



|  |
| --- |
| #cloud-config  ***package\_upgrade: true***  ***packages:***  ***- nginx*** |



#### REDEPLOYING THE VM

|  |  |
| --- | --- |
|  | 1. **RESET PASSWORD** – To Reset the VM password 2. **REDEPLOY AND REAPPLY –**  * If there is any sort of issue on the underlying physical server that is hosting   the VM. For example –   * If we can't log into the VM * Problem with the underlying physical host. * Maintenance on the underlying physical * Then we can click on redeploy. Redeploying the VM will deploy the VM to another physical host. |

SUBSCRIPTION QUOTA

* We do have a limit on number of VM we can create the number of VM we can create. The limitation is always tied to the type of subscription we have.
* Navigate to Dashboard 🡪 Subscription 🡪 Open the subscription 🡪 Usage and Quota



* Note – when we delete the resource group – this will delete all the resources in that resource group.

### STATE OF VM

Here we want to see the impact of “RESTARTING”, “STOPING” and “SHUTING DOWN” the VM

#### IMPACT ON TEMPORARY STORAGE

* The temporary disk is not recommended to store application data. ***The data in temporary disk will be lost if the VM is restated due maintenance of underlying physical server or VM itself is redeployed***
* We can attach a data disk to store application/personal data.
* When we do a restart from VM (after doing RDP) or a restart from the Azure portal itself. There will not impact on temporary storage. But – if we do a STOP from the Azure, the data in the temporary storage will be lost.
* **The reason behind this is – when we STOP the VM – Azure deallocates the VM from the underlying physical server and new location in the physical server is allocated after the restart.**
* *Note – We will not be charged for the compute cost of the VM if the VM is STOPPED (deallocated). But we will be charged for other aspects of VM like data disk*

IP-ADDRESS

* When we do a restart from VM (after doing RDP) or a restart from the Azure portal itself. The public IP will not change
* But – if we do a STOP from the Azure, the old public IP will be lost and the VM will get a new public IP after we Restart it back
* **The reason behind this is – when we STOP the VM – Azure deallocates the VM from the underlying physical server and new location in the physical server is allocated after the restart.**

## AVAILABILITY

### AVAILABILITY SETS

**WHY WE NEED AVAILABILITY SET?**

When we have the VM deployed to physical hardware. The VMs can be impacted due to

* **PLANNED MAINTENANCE ACTIVITY** - Due to periodic update on the physical hardware. In this case – azure will ask the user to restart the VM so that the VM get alloocted to new Physical hardware and get deallocated from the current physical hardware – which needs updates
* **UNEXPECTED DOWNTIME** - Due the power failure or networking issues
* To ensure the high availability – azure offers - “**AVAILABILITY SET**”

**AVAILIBILITY SET AND ZONES ARE CREATED TO INCREASE THE AVALIBILITY OF THE APPLICATION DEPLOYED TO VMs**

|  |  |
| --- | --- |
|  | * This feature helps to protect the machine/ application against the infrastructue level failure like network failure , local disk failures and even rack failures or it can be planned maintenance events – where microsoft updates the undelying phycial environment – where a reboot is required * We can increase the availability of the application by making use of avalibility set. Each VM that is assigned to a availibility set is assigned a separate Fault and update domain. * Availability Sets takes the virtual machine and configures multiple copies of it. **Each copy is isolated within a separate physical server, compute rack, storage units and network switches within a single datacentre within an Azure Region**. |

#### UPDATE AND FAULT DOMAIN

|  |  |
| --- | --- |
|  | **Availability set is a logical grouping of VMs. When we create a VM and make use of “availability set” the VMs. The VM are added to Fault and Update Domain**     * **FAULT DOMAINS**: Group of VMs sharing a common power source and network switch. We can create upto 3 fault domains * **UPDATE DOMAINS**: Group of VMs that are rebooted (updated) at the same time.We can create up to 20 update domains. |

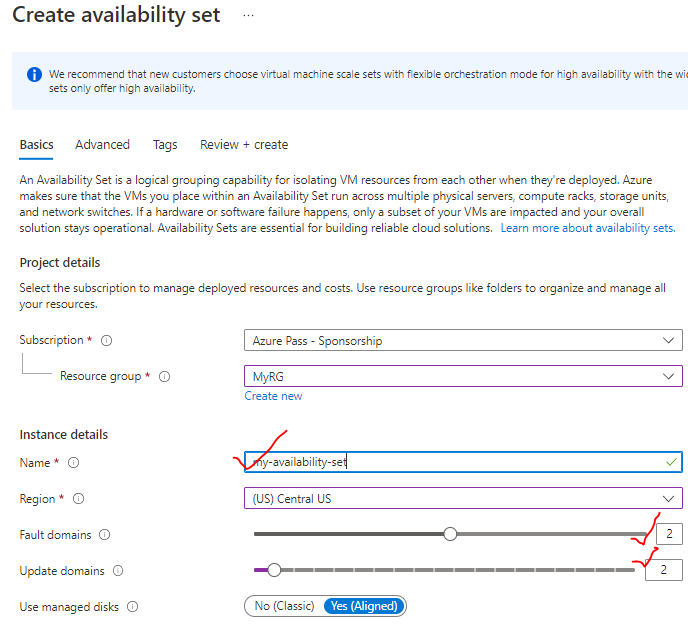
##### EXAMPLE

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **UD1** | **VM1** | **VM3** | | **UD2** | **VM2** | **VM4** | |  | **FD1** | **FD2** | | * In the following matrix diagram – Virtual Machine VM1 & VM2 belong to a Fault Domain (FD1)and VM3 and VM4 belong to fault domain FD2 * VM1 and VM3 belong to update domain UD1 and VM2 an VM4 belong to update domain UD2 * As the fault domain shares the common power source and network switch – Just in case some goes wrong with this fault domain – then the application will be still available via VM3 and VM4 and vice versa. * When the physical server needs an update – It will be updated based on update domain. Hence – if update domain UD1 is getting updated then application will be available via VM2 and VM4. |

##### DEMO

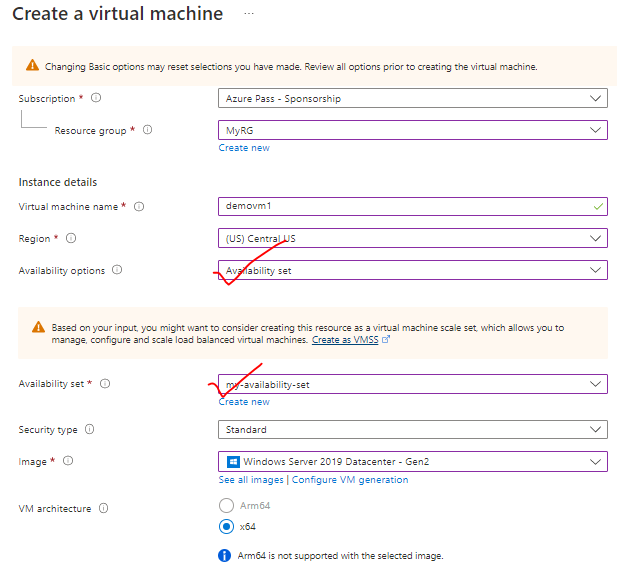
1. CREATE AN AVAILABILITY SET

* While creating availability set we are assigning 2 fault and 2 update domain
* Availability set can be assigned to a VM while creating the VM. Any existing VM cannot be made part of an availability set.

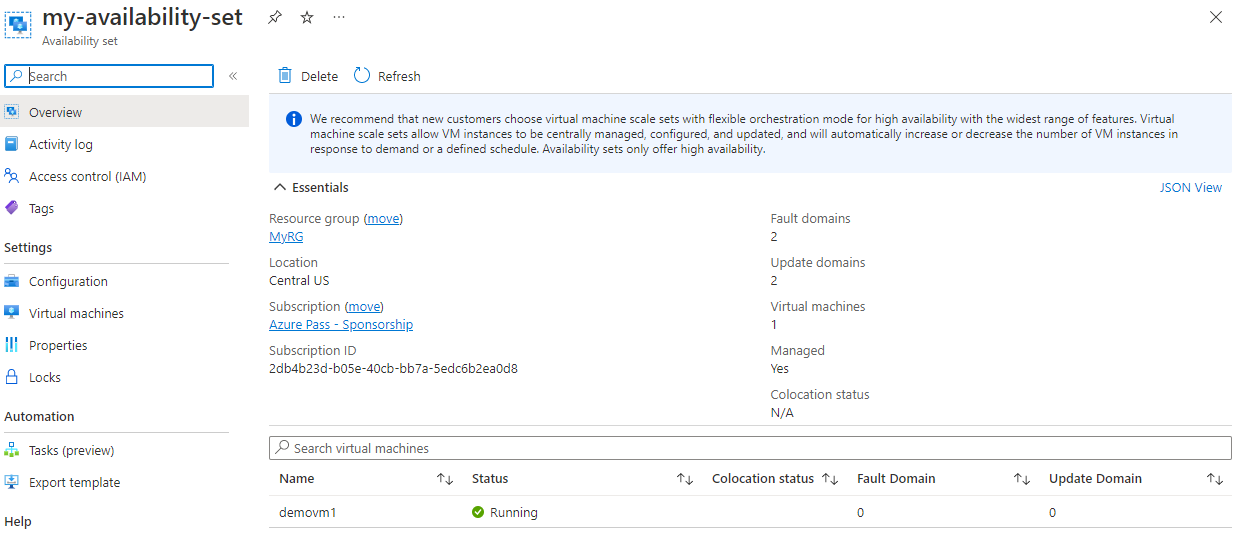


* Let’s create 2 VMs and make them a part of an availability set. Here Azure will make sure that VMs are properly distributed among fault and update domain.

*CREATING THE FIRST VM*



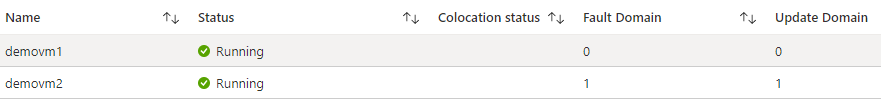
*AVAILABILITY SET STATUS:* The first VM is created in Fault Domain = 0 and update Domain= 0



*CREATING THE SECOND VM IN THE SAME AVAILABILITY SET*

* Follow the same Step to create the another VM in the same Availability Sey

*AVAILABILITY SET STATUS:* The second VM is created in Fault Domain = 1 and update Domain= 1



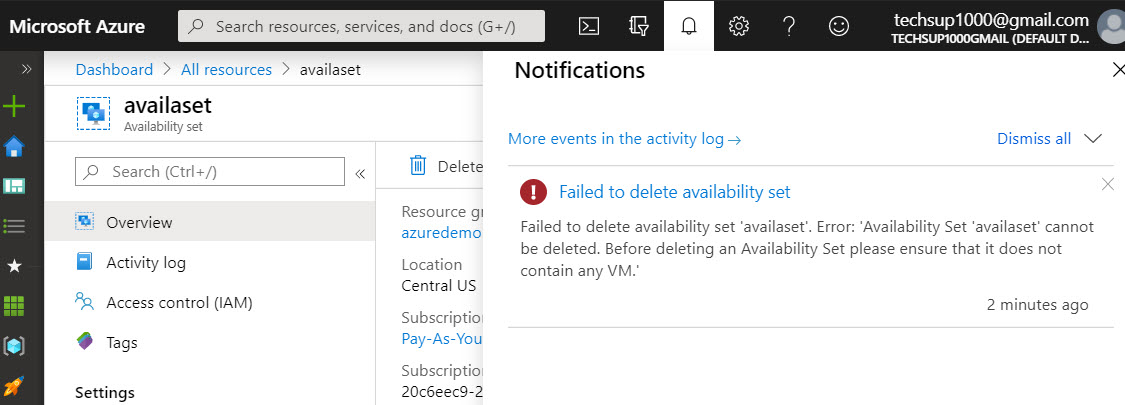
##### NOTES

* Azure makes sure that VMs are evenly distributed between the Fault and update domain in an availability zone.
* The configuration of availability set for VM can only be done while creating the VM.
* VM belong to another availability set cannot be migrated to. another availability set

**Note: The application deployed to a VM needs to be synced to another VM manually even it belongs to same availability set.**

**What happens if we try to delete the availability set as it is?**

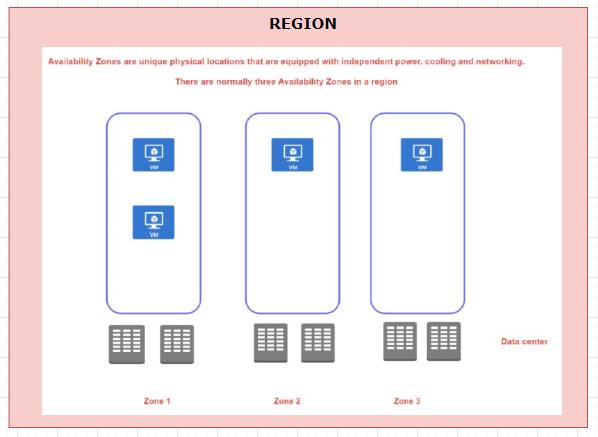
If we just try to delete the availability set as it is, we will get the following error message



Hence, we first have to ensure no virtual machine is associated with the availability set.So to delete the availability set, you can first delete the virtual machines linked to the availability set and then go ahead and delete the availability set.

### AVAILABILITY ZONES

* In a specific region – we have multiple zones and each zone is a collection of data centers.
* Availability zones are unique physical location that are equiped with independent power , colling and networking.
* This features help to provides better availability for the application by protecting them from datacenter failures
* Each Availability zone is a unique physical location in an Azure region
* Using Availability Zones, an availability of 99.99% is guranteed for the virtual machines.
* We need to ensure that we have 2 or more virtual machines running across multiple availability zones.



* When we create a VM in an availability zone – It is in turn mapped to the data center in that availability zone.
* The advantage we get with availability zone is that – if a data centers in a zone goes down – the application will be up and running from other zones in a specific region.

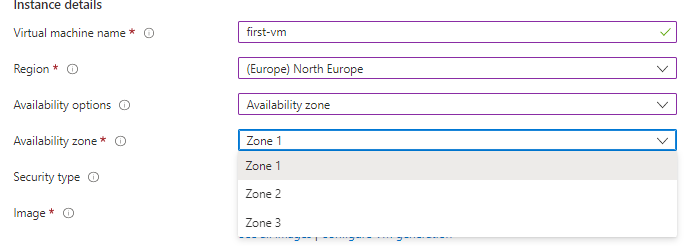
#### NOTES ON AVAILABILITY ZONES AND SET

* There is no extra cost involved in creating availability zone or sets. But there is a costing aspect – when it comes to VM communication between the Zones. But this cost is not applicable when it comes to availability set as those VM are part of same physical data center.
* When we create a VM is availability zone or set – it our responsibility to sync the application in all the newly created VMs- To achieve the sync we can make use of extension of custom script called “Cloud Init Script”.

#### EXAMPLE

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | * For a region we have availability zone. * Lets create 2 VMs in two different availabilty zones in a same region.  |  |  |  | | --- | --- | --- | | VM | REGION | AVAILABILITY ZONE | | VM1 | EAST US | 1 | | VM2 | EAST US | 2 | |

CREATING THE VM IN AVAILIBILITY



**VMS IN DIFFERENT AVAILIBILITY ZONE**

|  |  |
| --- | --- |
| VM1- OVERVIEW | VM2- OVERVIEW |
|  |  |

**WHY SHOULD I CREATE AVAILABILITY SETS IF WE HAVE AVAILABILITY ZONE?**

* When we create VM in different availability zone, there is an extra cost involved – which is called bandwidth pricing (<https://azure.microsoft.com/en-us/pricing/details/bandwidth/> ).
* Hence, we need to consider the bandwidth pricing while creating the VM is different availability zones.
* Bandwidth Price involve the charges on the data transfer between VM on different zones, especially when it comes to data used while syncing the application between VMs

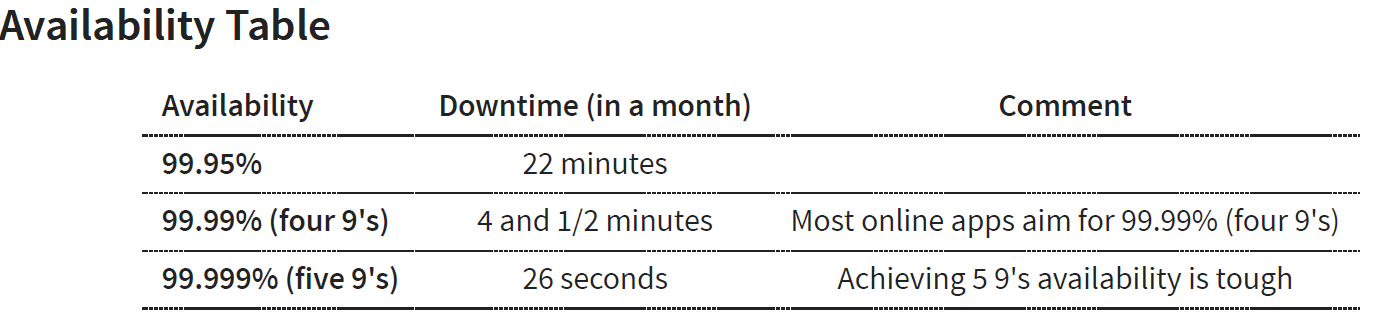
#### INCREASING THE AVAILABILITY

* **SINGLE INSTANCE VM:** If we are using single instance VM we can use a specific disk type . Below is the availbility – when we select a spefic disk type(from Disk Tab)

|  |  |
| --- | --- |
| **DISK TYPE** | **AVAILABILITY** |
| Premium SSD or Ultra Disk | 99.9% |
| Standard SSD Managed Disks | 99.5% |
| Standard HDD Managed Disks | *95%* |

Two or more instances in same Availability Set: 99.95%

* ***Summary: Create multiple instances in multiple AZs if you want high availability***



* Two or more instances in two or more Availability Zones in the same Azure region: 99.99%

## SCALE VMS IN AZURE

For high availability, scalability, and redundancy of VMs, azure has several features that can meet them. These features include:

* + VIRTUAL MACHINE SCALE SETS
  + AZURE BATCH

### WHAT IS AZURE BATCH?

Azure Batch enables large-scale parallel and high-performance computing (HPC) batch jobs with the ability to scale to tens, hundreds, or thousands of VMs. When you're ready to run a job, Batch does the following:

1. Starts a pool of compute VMs for you.
2. Installs applications and staging data.
3. Runs jobs with as many tasks as you have.
4. Identifies failures.
5. Requeues work.
6. Scales down the pool as work completes.

There might be situations in which you need raw computing power or supercomputer-level compute power. Azure provides these capabilities.

### VIRTUAL MACHINE SCALE SETS

* Let's say we have hosted an application which has some sort of workload that needs to be hosted on a virtual machine.
* If that the application is putting high load on the virtual machine- To can solve this issue one alternative can be
  + To change the size of the virtual CPU. but there is a limit on how much we can scale the VM. And sometimes the application might not behave properly just by increasing the number of virtual CPUs or the memory.
  + Another alternative to this is to host the application on multiple VMs and then we increase or scale the number of similar VMs.

**WHAT IS THE ROLE OF VM SCALE SET?**

* The VM scale set service we can provision identical virtual machines. The provisioning of VM scale set is done on demand.
* VM scale sets can be used for static application i.e., the application that starts with a specific state always. (Similar to calculator – which always start from “0” when its starts)

EXAMPLE

* Let’s say for an application- we have one machine that has been created by the virtual machine scale set. Now- we can define rules. Saying - scale out if the CPU percentage goes beyond 70%.
* Hence - If the CPU utilization goes beyond 70%, then add more VM. So eventually - we'll have two machines as part of our infrastructure.

#### SCALLING IN VM SCALE SET

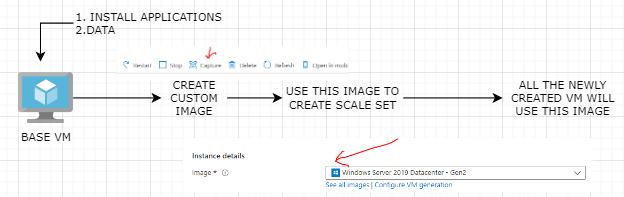
1. At peak demand /load. We can have three machines that have been spun up. If we have three machines, we are billed for all three machines
2. But when the demand of the application has become less. According to the scale set rule (scaling policy) on scale set it will be scaled in, let’s say 1 VM – Then we will be charged for just for 1 VM.
3. In the scaling rule we define the minimum and maximum number of VM during scale in and out respectively

#### SET UP OF VM SCALE SET

|  |  |
| --- | --- |
|  | * All the VM in the scale set are integrated with a Load Balancer. The configuration of add a LB can be done while create a VM scale set. * The load balancer has a public IP. When the user hit the public IP – the request will go the load balancer * In the load balancer we configure some load balancing rules – to govern how the load balancing has to be done * When VMs get added to the scale set – it will also get integrated with load balancer and get removed from LB whenever the VM is removed from the scale set. |

HOW THE APPLICATION / DATA GETS INSTALLED /COPPIED TO THE NEW VM, WHEN SCALE OUT HAPPENS?

* This is done using the custom image
* Step 1: We first prepare a base VM – where we install all the applications
* Step 2: We create a custom image from the Base VM
* Step 3: Now this custom image is used to create the VM scale set. Hence with the scale out – the VM will use the same image to create identical VM



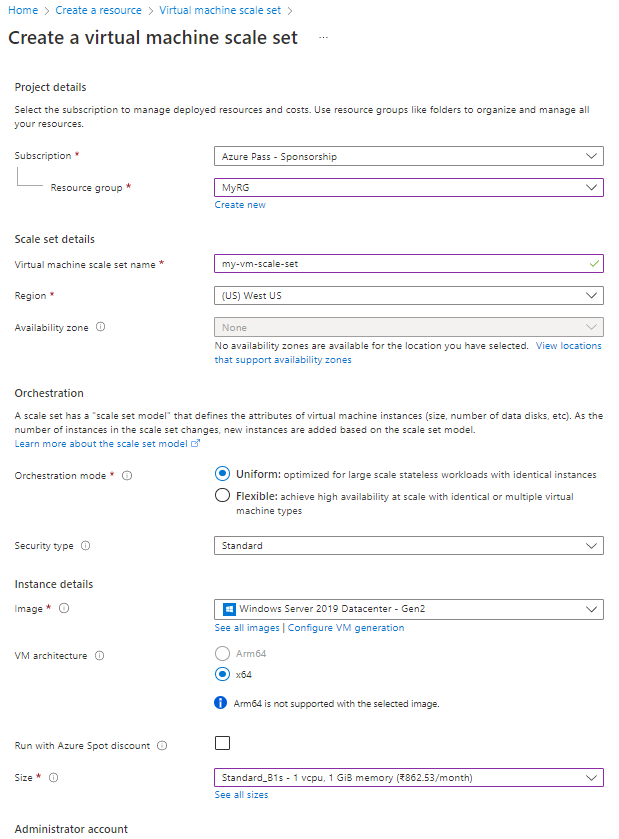
#### CREATING A VM SCALE SET

1. Create a resource of type VM Scale set



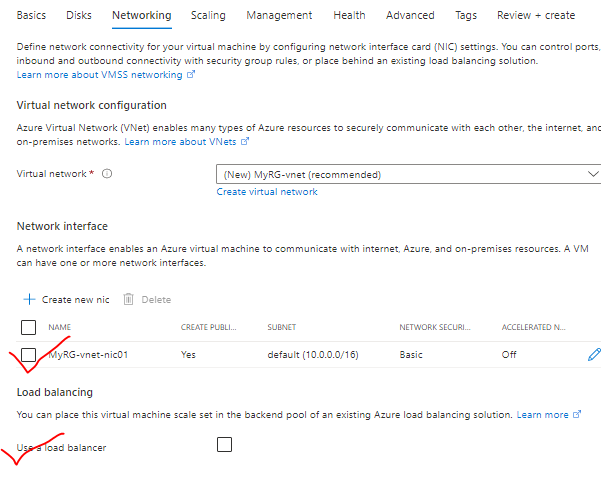
1. Configure the VM scale set – All the configuration below is the configuration - **that's going to be used as the underlying image for the different instances that are going to be part of the virtual machine skill set**

##### BASIC TAB

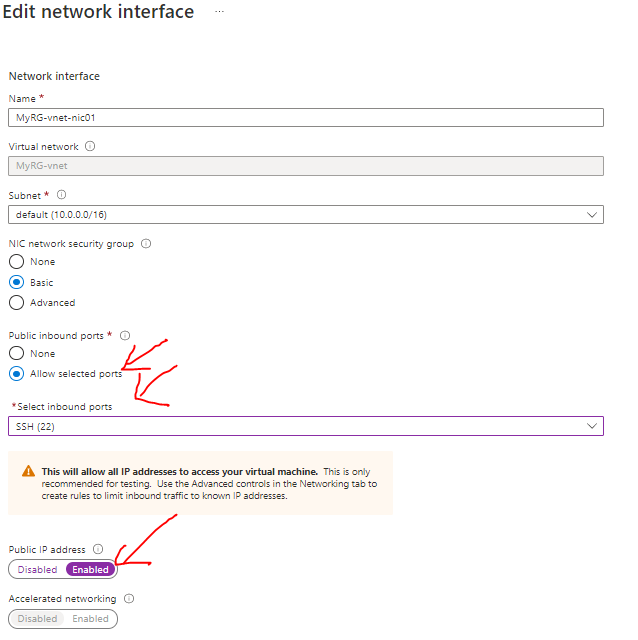


##### NETWOKING TAB

* Edit the Network interface
  + Allow the SSH 22 port so that we can be able to do SSH to the machine
  + Enable the public IP for the scale set
* We can attach a load balancer to the VM scale set – which can distribute the load across the VMs when it is scale out.



###### NIC CARD CONFIGURATION

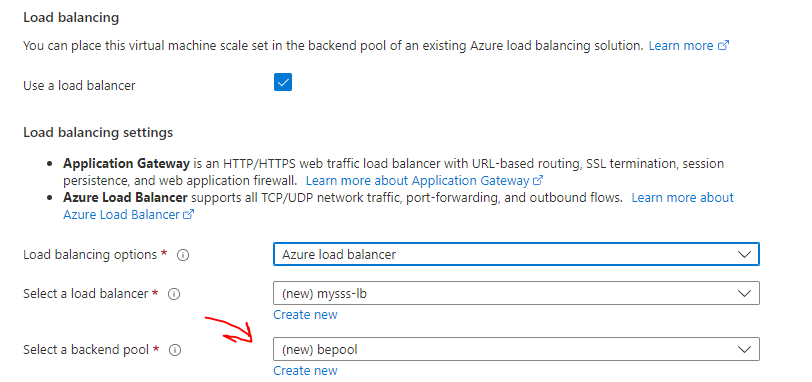


###### LOAD BALANCER

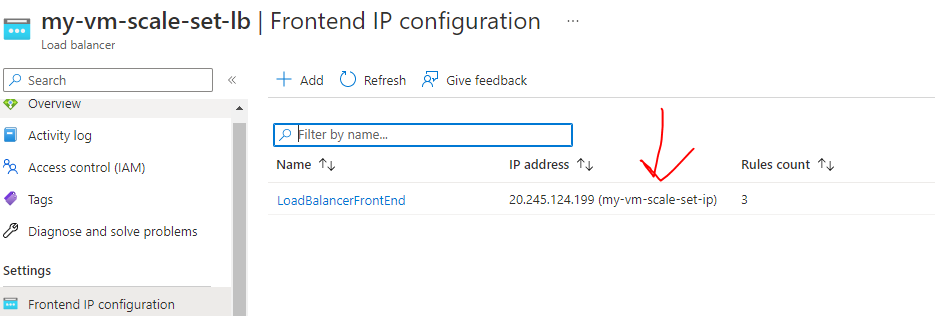
* When a load balancer is configured for a scale set a backend pool is created

WHAT IS BACKEND POOL?

* A backend pool is set of VM to which load balancer will redirect the request.

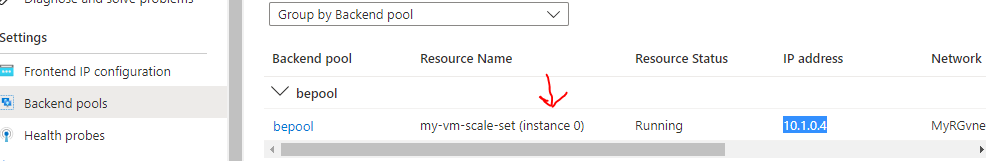


PUBLIC IP OF LB

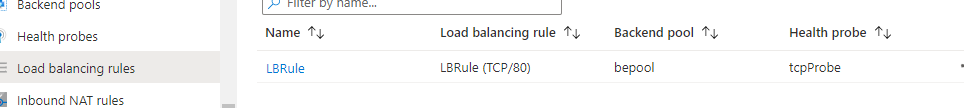


BACKEND POOL OF LB

* It is mapped to the VM of the scale set so that the request can be routed to the VMs

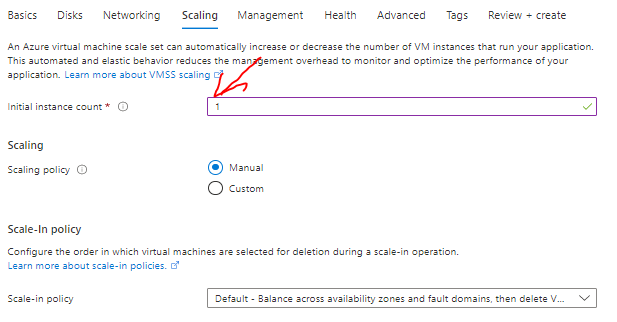


LB RULES



##### SCALING TAB

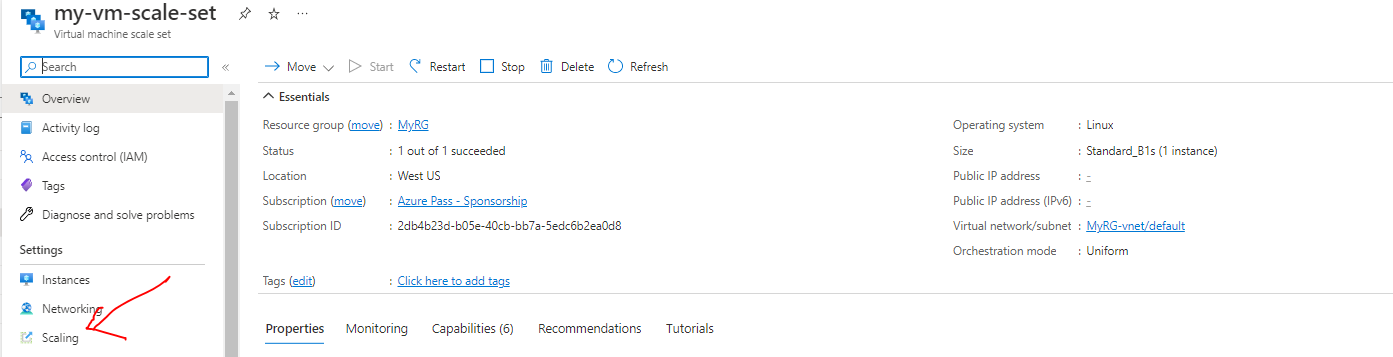
* Here we configure number of VM instance that will be created initially in the scale set.
* ***The maximum number of identical VM that can be provisioned in a scale set is 1000 when we use market place image. For custom images it is limited to 600 VMs***
* In case of Manual scaling – we need to manually update the number of VM instance when the load increases or decreases from the scale set service.



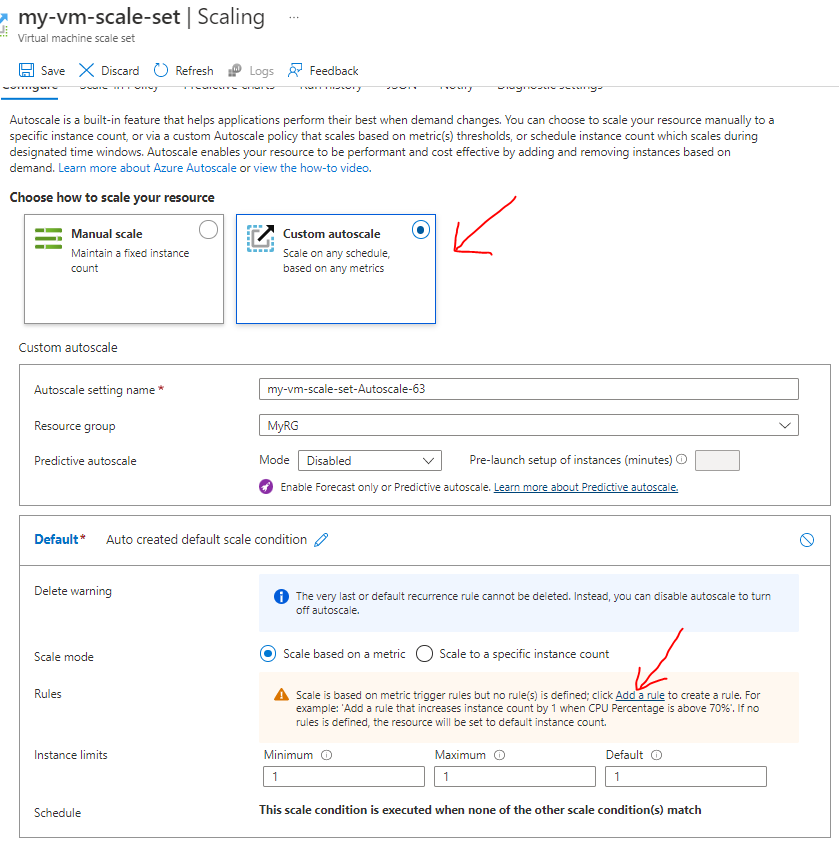
* Then Review & create 🡪 Create

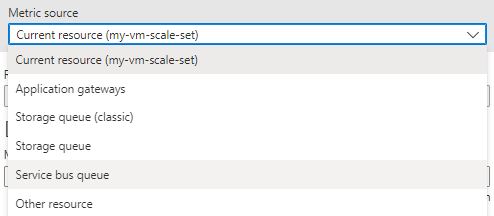
###### ADDING A SCALING RULE TO SCALE SET

* Once the scale set is created – we need to attach rules to the scale set to scaling -in and Scale-out the VM i.e., in what condition a new identical VM will provisioned once the load increases and scale down when the load decreases.

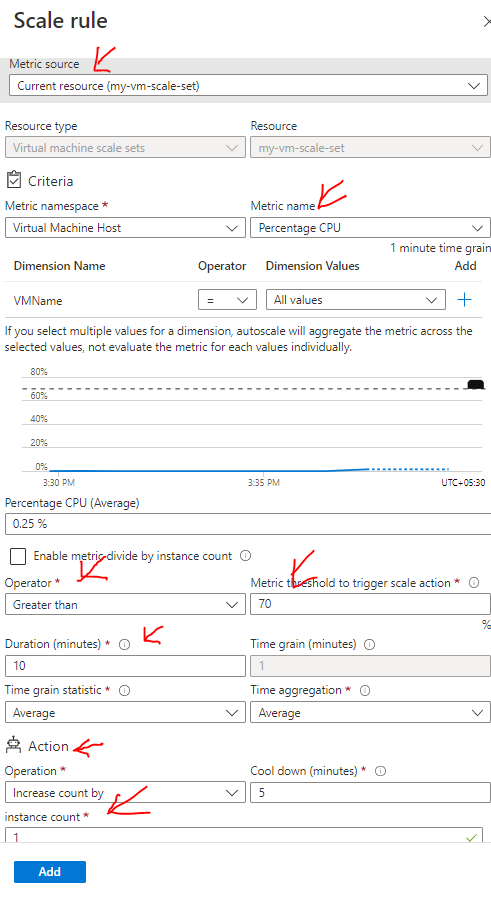


* Go to the scale set resource 🡪Scaling 🡪 custom Auto Scale to set the scaling rule
* To scale the Scale set based on metric 🡪 Select “Select based on metric”🡪Click to “Add a Rule” link to add scale set rule.

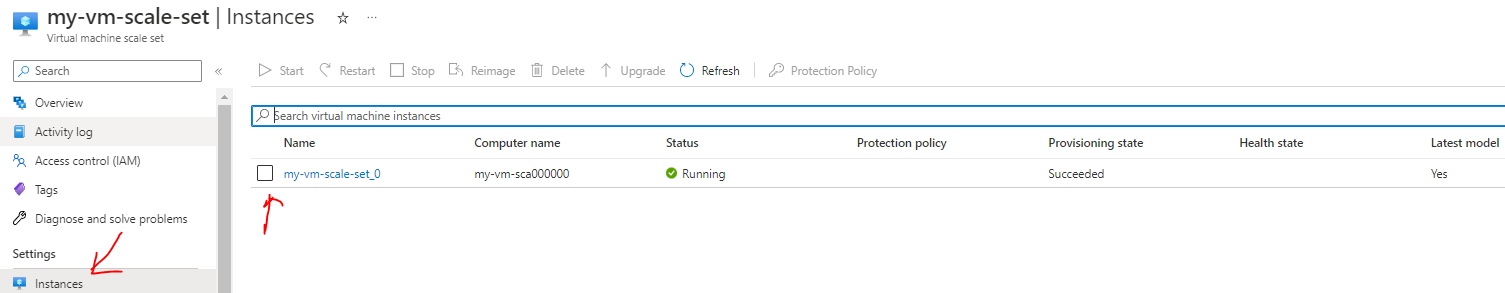




* The below scale set rule says - In the VM scale set *(Metric Source*) if the CPU utilization (*Metric Name*) increases more than 70% then increase the VM by 1(*instance count*)
* Click on Add 🡪 Save



* In the instance setting - we can see number the number of instances in the scale set
* If the CPU utilization increases beyond 70% - we can see a greater number of VM getting added to the list below.



#### SIMULATION

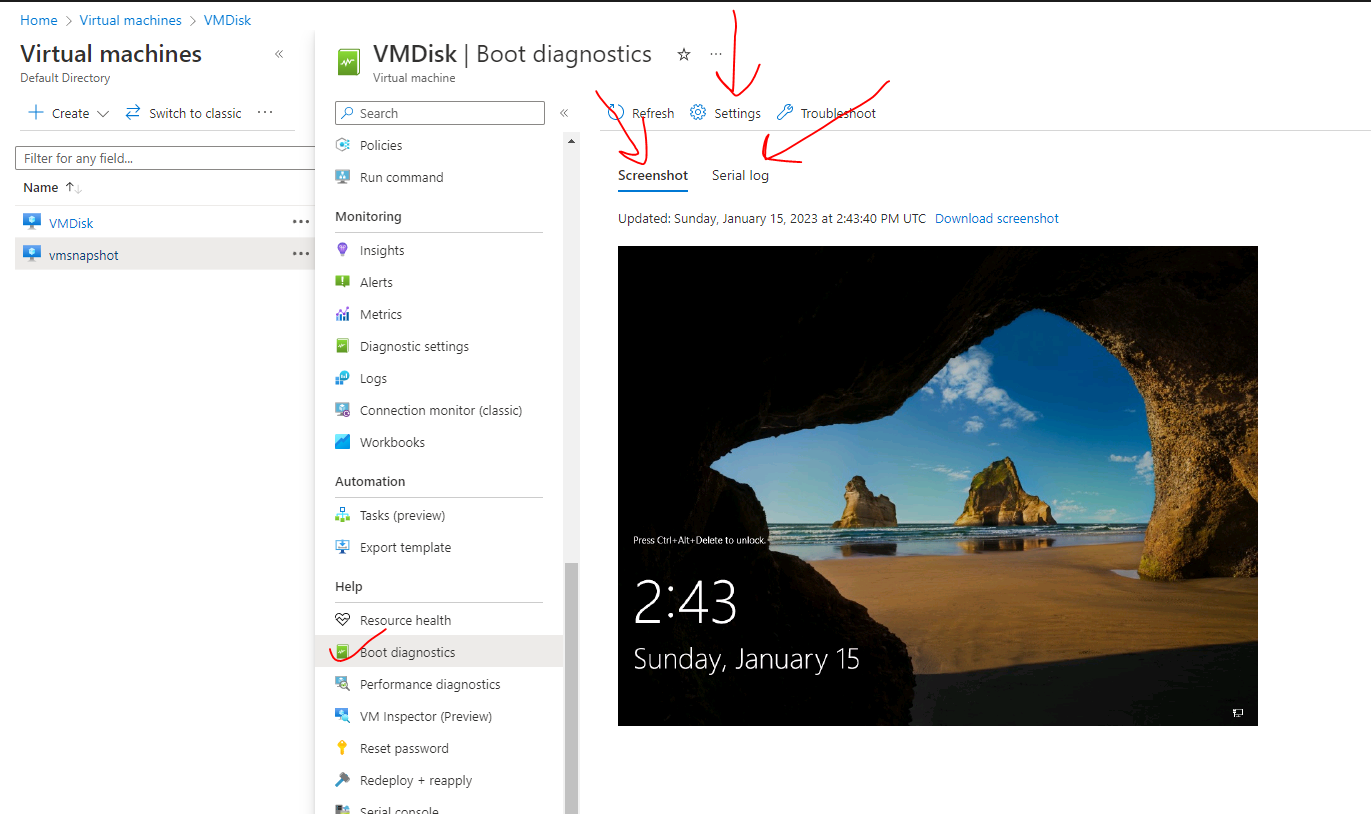
* To simulate the “high CPU” utilization use case and trigger the Scaling rule – We can login to the VM using its Public IP
* Click on the above VM 🡪 copy the public IP 🡪 Login to VM using putty
* To simulate it using a tool called “***stress***”

#### CONNECTING TO A VM SCALE SET

Details to be added

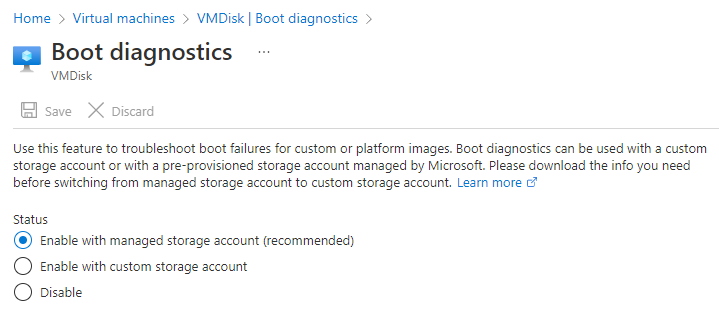
## VM -BOOT DIAGNOSTICS

* <https://learn.microsoft.com/en-us/azure/virtual-machines/boot-diagnostics>
* Boot diagnostics is a debugging feature for Azure virtual machines (VM) that allows diagnosis of VM boot failures. Boot diagnostics enables a user to observe the state of their VM as it is booting up by collecting serial log information and screenshots.



### VM -BOOT DIAGNOSTICS SETTINGS

* Note – The data of the boot diagnostic are stored in the storage account

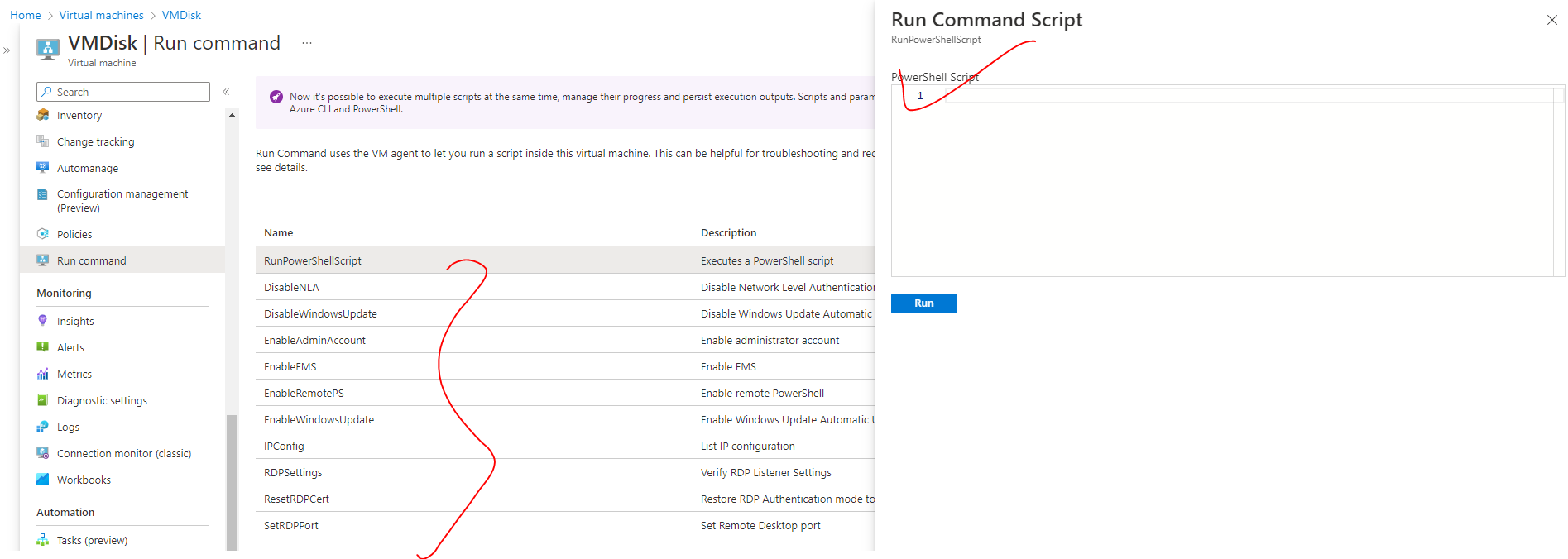


## VM- SERIAL CONSOLE

* Serial consoles are used to send command to VMs. It opens a console (similar to command prompt) – which can be used go run command on the VM.

|  |  |  |
| --- | --- | --- |
| STEP-1 | **cmd** |  |
| STEP-2 | **ch -sh 1** |
| STEP-3 | Enter credential |
|  | |  |

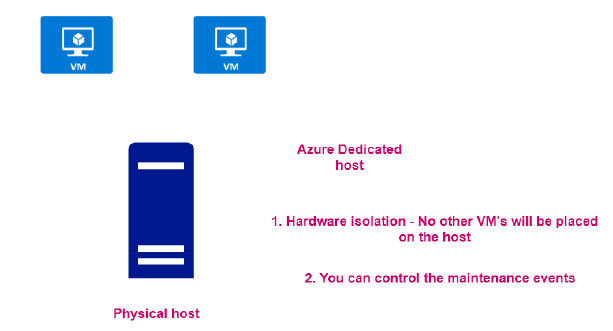
## VM – RUN COMMAND



* In Run command option – It has multiple VM agents that allows us to run a script inside the virtual machine itself. It has different options like PowerShell etc.

|  |  |
| --- | --- |
| EXAMPLE – Lets say we need to install IIS web- server we need to run following Powershell command | import-module servermanager  add-windowsfeature web-server -includeallsubfeature |

## AZURE DEDICATED HOST



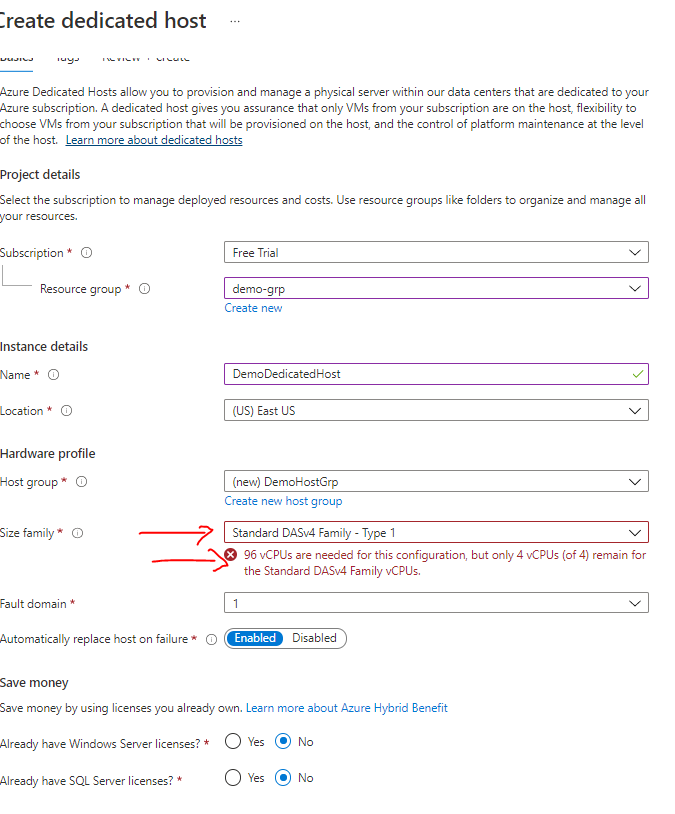
* When we create VMs, they are get created in physical server/ host in the azure data center
* In Azure – it come with a capability to assign the entire physical host as a resource. Note this capability are usually leveraged by large scale organization.

### ADVANTANGE OF DEDICATED HOST

* As it is a dedicated host – no other VM can be placed in the host.
* We can be able to control the maintenance events of the VMs

### CREATING DEDICATED HOST

* Navigate to Market Place 🡪 Search “Dedicated Host”.
* The creation of Dedicated host failed in below diagram as it need more resources (i.e 96 CPU) – which are not applicable for free subscription.

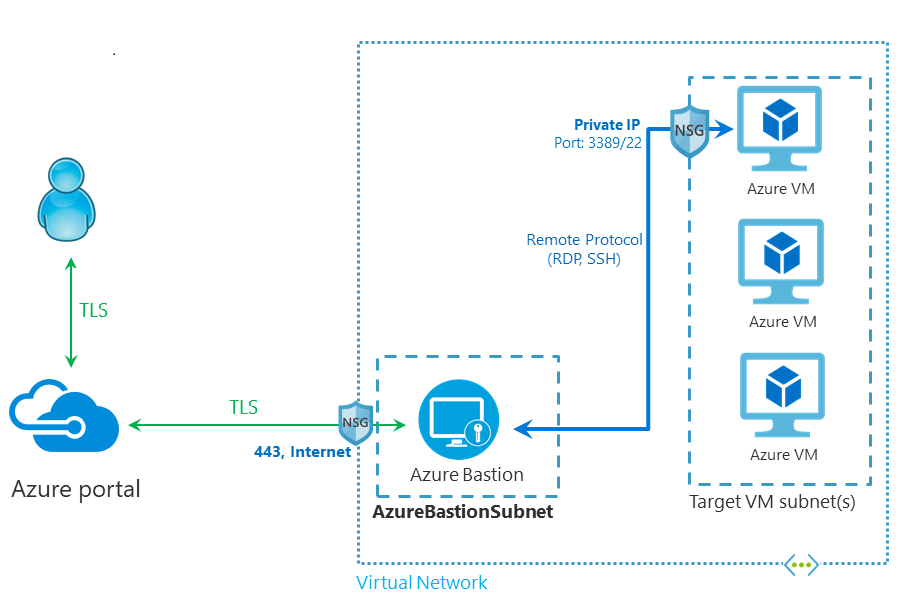


## AZURE BASTION SERVICE

1. It an PAAS offering from Azure.
2. This provides seamless RDP and secure shell connectivity to Windows and Linux VM respectively. The connection can be directly from Azure portal directly.

### USE CASE

* In a virtual network, we might have VMs that are part of different subnets – we can connect to the VM using the public IP of the VM
* But in cases -where we are running internal workloads and we do want to expose them to the public internet. Hence no public IPs are assigned to those VMs. In such cases we can make use of Bastion Services to connect with those VMs and perform admin tasks. This gives an extra level of security on the VMs



* The connection to be done via Port 443 and we can connect onto the virtual machines that are part of the virtual network.
* ***When we use this service, it's going to create its own compute machines(Virtual Machine Scale set) which is responsible for establishing that connectivity. All of this is going to be managed in the background. All these machines are managed by the Azure bastion service are going to be launched in a separate subnet in the virtual network.***
* Both the Bastion Subnet and the Target VM Subnet must be part of same VNET. The name of the bastion subnet must be **AzureBastionSubnet.**
* **The Azure Bastion is basically Virtual machine in the background**

### SETTING UP BASTION SERVICE

|  |  |
| --- | --- |
|  | Step 1:   1. Create a VM to which we will be connection via Bastion service. 2. If the VM has associated public IP, then disassociate the Public IP address - since we will be using bastion service to connect with the VM rather than its public IP. |

#### DISASSOCIATING THE PUBLIC IP FROM VM

|  |  |
| --- | --- |
|  | * The dissociation of public IP address can be done while provisioning the VM itself. * These steps are for dissociation of public IP after the VM is created * We can delete the public IP address too since public IPs are separate resource in Azure. |

|  |  |
| --- | --- |
|  |  |

#### SETTING UP BASTION

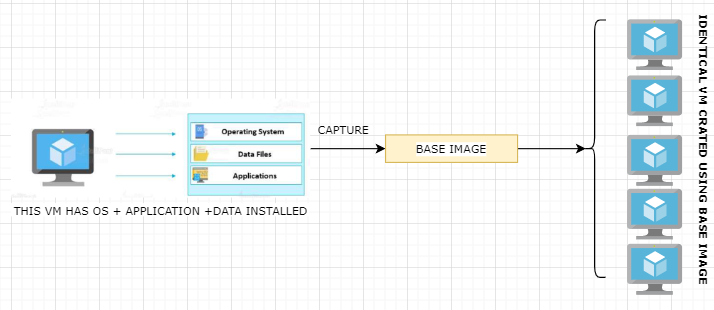
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Search for Bastion Service 🡪 Create bastion  **TIER:** Note the bastion service is VM in background   |  |  | | --- | --- | | **STANDARD** | **BASIC** | | Basic supports max 50 instances of VM | Basic supports 2 instances of VM |   INSTANCE COUNT   * ***Each instance can support 20 concurrent RDP connections and 40 concurrent SSH connections for medium workloads.*** * Note - The number of connections per instances depends on what actions you are taking when connected to the client VM. For example, if you are doing something data intensive, it creates a larger load for the instance to process. Once the concurrent sessions are exceeded, an additional scale unit (instance) is required. * To configure the **AzureBastionSubnet**, select Manage subnet configuration. * On the Subnets page, select +Subnet to open the Add subnet page. * On the Add subnet page, create the **'AzureBastionSubnet'** subnet . Note -The subnet name must be **AzureBastionSubnet**. * ***The subnet must be at least /26 or larger (/26, /25, /24 etc.) to accommodate features available with the Standard SKU.*** * Hence to support for maximum supported instance in Standard SKU – we need 50+5 = 55 IP address. That’s why subnet must have minimum 55 available host .  |  |  |  | | --- | --- | --- | | 10.0.1.0/26 | Default Subnet Mask | 255.0.0.0 | | Subnet Mask | 255.255.255.192 | | Total Number of network 2^(number of bits borrowed from Host bits) | 2^18= 262144 | | **Total Number IP address 2^(number of host bits)** | **2^6= 64** | | Number of Hosts | 2^6-2 = 62 | |
|  | |
|  | | |
|  | | |
|  |  | |

* We can also make use of wizard go deploy bastion
* Go the VM 🡪 Bastion 🡪 Deploy Bastion

|  |  |
| --- | --- |
|  |  |

## VIRTUAL MACHINE - CUSTOM IMAGES

* An image is a copy of a VM or a template for creating a VM , which might contain an OS , data files and applications. Using custom images we can create VMs.
* For example – Let’s say we have a VM which has some data and application installed. To provision identical VM which will have same data and applications already installed
* For such scenario – we create a base image from the VM which then used to create identical VMs.



* After creating the image – we can place the image in Azure compute gallery.
* The Azure compute gallery can be shared – so that another user can use the image to create a VMs

### TYPES OF CUSTOM IMAGES

1. SPECIALIZED VM IMAGES
2. GENERALIZED VM IMAGES

|  |  |
| --- | --- |
| **SPECIALIZED VM IMAGES** | **GENERALIZED VM IMAGES** |
| Information about specific users and machine information is retained | Information about specific users and machine information is removed |
| VM created using Specialized VM images will have same computer name and admin user information | Here we perform a process of generalization. The original VM is unusable after we perform generalization. |

### CREATING A SPECIALIZED IMAGE

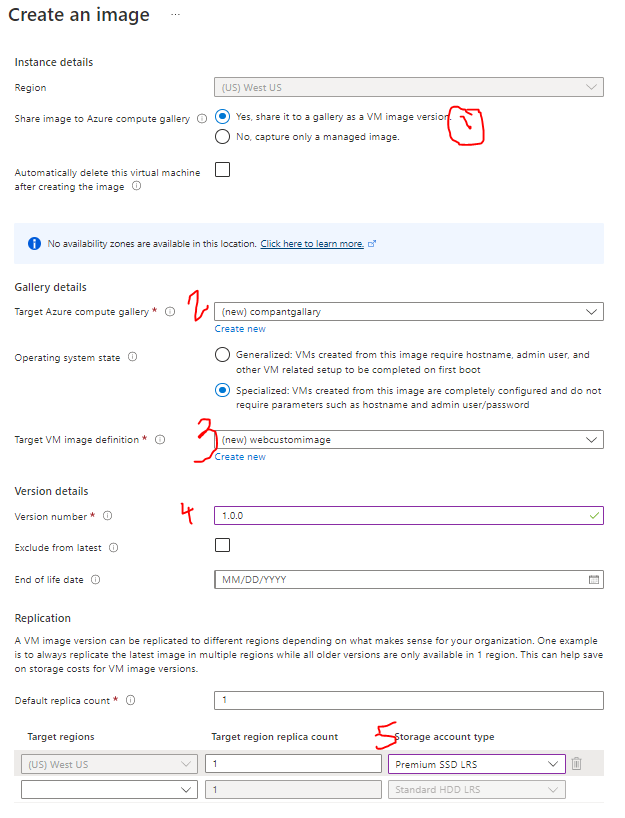
Let say we have a VM which has following applications and data in place

* IIS installed (with custom welcome page)
* Data disk attached
* Data file (txt file) in the data disk

To create a specialized image out of this VM 🡪 Go to the VM 🡪 Capture

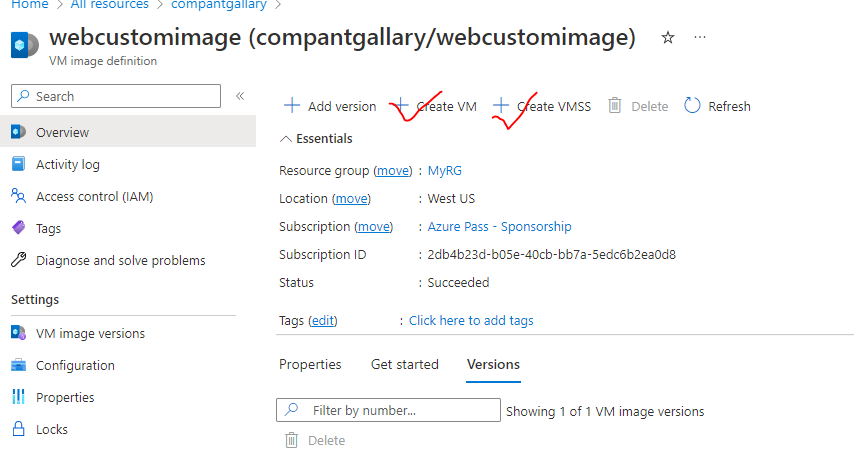
|  |  |
| --- | --- |
| 1 | Image can be   1. **PART OF AZURE COMPUTE GALLERY** 2. **MANAGED IMAGE** –    1. To create a managed image, we need to generalize the underlying VM. |
| 2 | Name of the compute gallery |
| 3 | Name of the image |
| 4 | Version Number of the image |
| 5 | Storage for the image |

* In the creation process of the custom image – the VM is first stopped and then a snapshot of the VM is taken
* Note – We should create image – we have to make sure all the process of VM or the VM is in stopped state
* When we create a specialized VM from a base VM. There will be no impact of the Base VM.

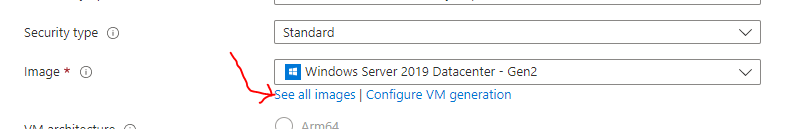


#### CREATING A VM / VM SCALE SET FROM THE CUSTOM IMAGE (SPECIALIZED)

* We can create VM or VM scale set using the custom image



FROM CREATE VM OPTION





#### CREATING A GENERALIZED IMAGE

* When we do generalization of a VM – the base VM becomes unusable
* To generalize the VM – We need to start the system preparation tool 🡪 Run Command🡪 type ***sysprep***

|  |  |
| --- | --- |
|  |  |
|  | * This will shutdown the VM * *To create image 🡪 Capture* |

* This will remove all the account and machine details – but the application and data are intact in the image