Contents

[CLOUD CONCEPTS 3](#_Toc87443851)

[IP ADDESSS 3](#_Toc87443852)

[RESOURCE GROUP 5](#_Toc87443853)

[CREATING A RESOURCE GROUP 5](#_Toc87443854)

[REGION AND ZONES 6](#_Toc87443855)

[SELECTING A REGION 7](#_Toc87443856)

[VIRTUAL MACHINE 7](#_Toc87443857)

[AZURE’S VIRTUAL MACHINE SERVICE 7](#_Toc87443858)

[AZURE VIRTUAL MACHINE – KEY CONCEPTS 8](#_Toc87443859)

[DEPLOYING A VIRTUAL MACHINE 8](#_Toc87443860)

[DEPLOYING WINDOWS VM 9](#_Toc87443861)

[DEPLOYING LINUX VM 14](#_Toc87443862)

[INSTALLING SOFTWARES IN VM 15](#_Toc87443863)

[QUOTA OF VM 16](#_Toc87443864)

[AZURE MARKET PLACE 17](#_Toc87443865)

[AVAILABILITY 18](#_Toc87443866)

[INCREASING THE AVAILABILITY 18](#_Toc87443867)

[AVAILABILITY SETS 18](#_Toc87443868)

[AVAILABILITY ZONES 22](#_Toc87443869)

[NOTES ON AVAILABILITY ZONES AND SET 23](#_Toc87443870)

[VIRTUAL MACHINE SCALE SETS 23](#_Toc87443871)

[AZURE DEDICATED HOST 24](#_Toc87443872)

[AZURE SPOT INSTANCES 25](#_Toc87443873)

[WORKLOAD 25](#_Toc87443874)

[IMPORTANT USE CASES AND SOLUTIONS 26](#_Toc87443875)

[AZURE VIRTUAL NETWORK 26](#_Toc87443876)

[CREATING A VIRTUAL NETWORK 26](#_Toc87443877)

[CREATING A SUBNET IN VIRTUAL NETWORK 28](#_Toc87443878)

[CREATING A VIRTUAL MACHINE IN VIRTUAL NETWORK 28](#_Toc87443879)

[COMMUNICATION ACROSS VIRTUAL MACHINES IN A VIRTUAL NETWORK 29](#_Toc87443880)

[NETWORK SECURITY GROUP 29](#_Toc87443881)

[APPLICATION SECURITY GROUP 30](#_Toc87443882)

[NETWORK CONNECTIVITY OPTIONS 31](#_Toc87443883)

[VIRTUAL NETWORK PEERING 31](#_Toc87443884)

[POINT-TO-SITE VPN CONNECTION 31](#_Toc87443885)

[SITE-TO-SITE VPN CONNECTION 32](#_Toc87443886)

[AZURE EXPRESS ROUTE 32](#_Toc87443887)

[AZURE CORE SERVICES 33](#_Toc87443888)

[AZURE STORAGE 33](#_Toc87443889)

# CLOUD CONCEPTS

## IP ADDESSS

* IP is the unique identifier of a device in a network.



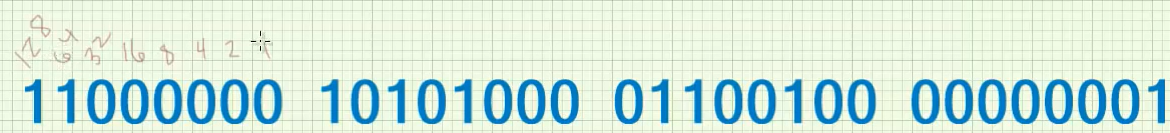
|  |  |
| --- | --- |
|  | * The IP address has 2 parts – The Network part and a Host part. * Each host in a network will have same Network Address |

#### IPv4 ADDRESSING

* It’s a 32-bit logical address
* It consists of 4 octet – and each octet ranges from 0 -255
* IP address has parts ***Network ID and Host ID***



#### BINARY TO DECIMAL



* When the IP is represented in Binary the decimal representation can be done using the ***power to 2***
* Hence the equivalent decimal will be 192.168.100.1 (128+64).(128 + 32 +8).(64+32+4).(1)

#### CLASSES IN IP ADDRESSING



* Ranges 127.x.x.x are reserved for the [loopback or localhost](https://www.computerhope.com/jargon/l/locahost.htm), for example, 127.0.0.1 is the loopback address.
* Range 255.255.255.255 [broadcasts](https://www.computerhope.com/jargon/b/broadcas.htm) to all hosts on the local network.

1. HOW TO DECIDE THE CLASS OF IP ADDRESS?

To decide the class of IP address we consider the first octet for example – **132**.20.10.192 – This IP belong to CLASS B.

#### NETWORK ID IN IP ADDRESSING

|  |  |
| --- | --- |
|  | * In Class A IP address 1st 8 bits are Network Id * Remaining 24 bits are for Host hence for class a we can have * Hence number of Host is **224 = 16777216** |
| * In Class B IP address 1st IP 16 bits are Network Id * Remaining 16 bits are for Host hence for class a we can have * Hence number of Host is **216 = 65536** |
| * In Class C IP address 1st IP 24 bits are Network Id * Remaining 8 bits are for Host hence for class a we can have * Hence number of Host is **216 = 256** |

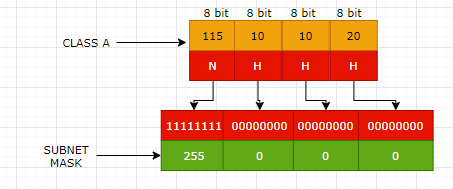
* Example : Find the network id of IP : **192.10.10.10** ?
  + Step 1 : The class of this IP is Class C
  + Step 2: The class B IP has first 24 bit (1st 3 Octet) as its network id . Hence the Network id : **192.10.10.0**

#### SUBNET, SUBNET MASK AND CIDR NOTATION

* Note the network id is represent by “1” and host is represented by “2”.

##### CALCULATING SUBNET MASK

* Example : Calculate the Subnet mask of : **115.10.10.20 ? Ans – 255.0.0.0**



###### WHAT IS THE JOB OF SUBNET MASK?

* Since the IP is consist of 2 part Network and Host part. Network Part in a network will be same for all host.
* Host decide the network Id with the help of Subnet Mask.
* Subnet mask is also of 32 bits -which has a mapping with the IP address. The 1s represent the network portion and 0s are the host portion. In the above example – When an IP is given to a device – then the subnet mask is also configured. The bits represented with “1” is the network id in the IP – when compared from left to right.

##### BROADCAST ID

* Broadcast IP is used to broadcast to all the host in the network.
* Find the class, network id ,broadcast id and usable IP of the following IP

|  |  |
| --- | --- |
| 150.10.20.30 | * The IP belongs to Class B * Network Id: 150.10.0.0 * Broadcast ID: To calculate the broadcast id, set the host part of Network to 255. Hence the broadcast Id - 150.10.255.255 * Usable Host IP = Total Number of IP address – (Network IP + Broadcast IP) i.e.   + For a network there will be on Network IP and one Broadcast IP.   + Usable Host IP = Total IP -2= 216 – 2 = 65536 – 2= **65534** |

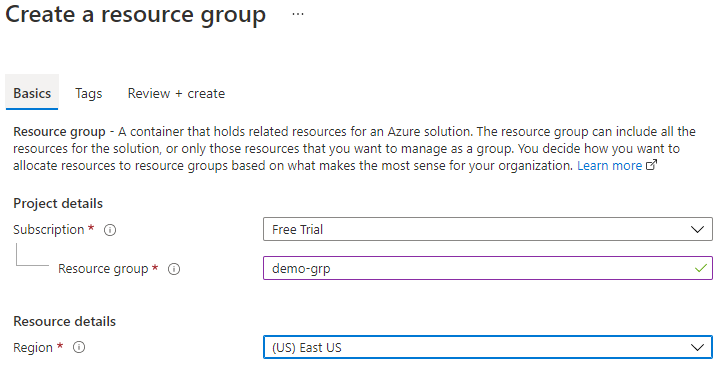
###### CIDR NOTATION (CLASSLESS INTER DOMAIN ROUTING)

* In class world we use subnet mask to deduce the number of bits used for Network
* Similarly, rather than using the subnet mask - CIDR notation is a way to represent the network.
* For example, **192.168.100.1/24**
  + This represent that the first 24 bits are used for network id. It is also called network pre-fix

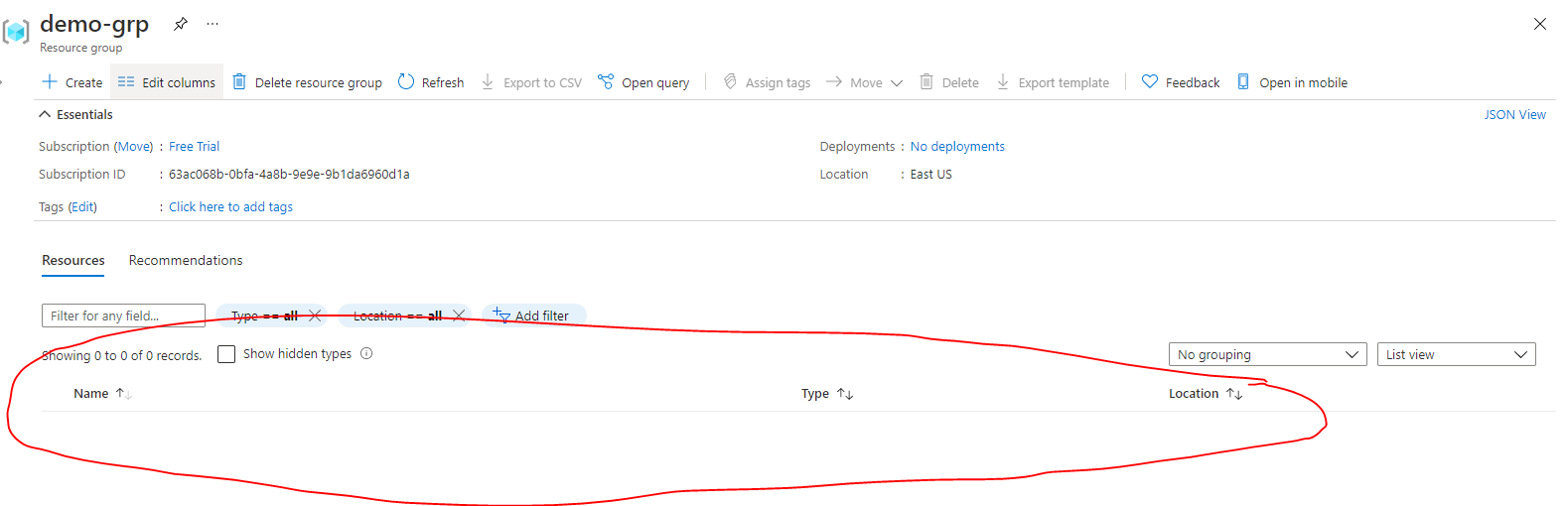
# RESOURCE GROUP

* Resource Group can be considered as a logical container/separation of resources. For example – For an organization resource group can be separated based resources a department of an organization is using.
* Every resource we create in Azure platform must be a part of resource group.
* There is no cost involved in creation of resource group.

## CREATING A RESOURCE GROUP



* All the resources are tied to a subscription for billing aspects.
* The resource group must associate to a region.
* Initially there will be no resource in the resource group. When we start adding resources to the resource group it will show up in the below table
* ***A resource cannot be a part of two different resource group***
* ***The resource can only be a part of a one subscription.***



# REGION AND ZONES

|  |  |
| --- | --- |
|  | CASE 1: SINGLE DATA CENTER IN A REGION  Imagine that your application is deployed in a data center in London  WHAT WOULD BE THE CHALLENGES?   * Challenge 1: Slow access for users from other parts of the world (high latency) * Challenge 2: What if the data center crashes?   *Your application goes down (low availability)* |
|  | CASE 2: MULTIPLE DATA CENTER IN A REGION  Let's add in one more data center in London  WHAT WOULD BE THE CHALLENGES?   * Challenge 1: Slow access for users from other parts of the world * Challenge 2 (SOLVED) : What if one data center crashes?   *Your application is still available from the other data center*   * Challenge 3: What if entire region of London is unavailable?   *Your application goes down*  16 |
| CASE 3: MULTIPLE REGION – MULTIPLE DATA CENTER    Let's add a new region: Mumbai  WHAT WOULD BE THE CHALLENGES?   * Challenge 1 (PARTLY SOLVED): Slow access for users from other parts of the world   *You can solve this by adding deployments for your applications in other regions*   * Challenge 2 (SOLVED) : What if one data center crashes?   *Your application is still live from the other data centers*   * Challenge 3 (SOLVED) : What if entire region of London is unavailable?   *Your application is served from Mumbai*  ADVANTAGES:   * *High Availability* * *Low Latency* * *Global Footprint* * *Adhere to government regulations* | |

## SELECTING A REGION

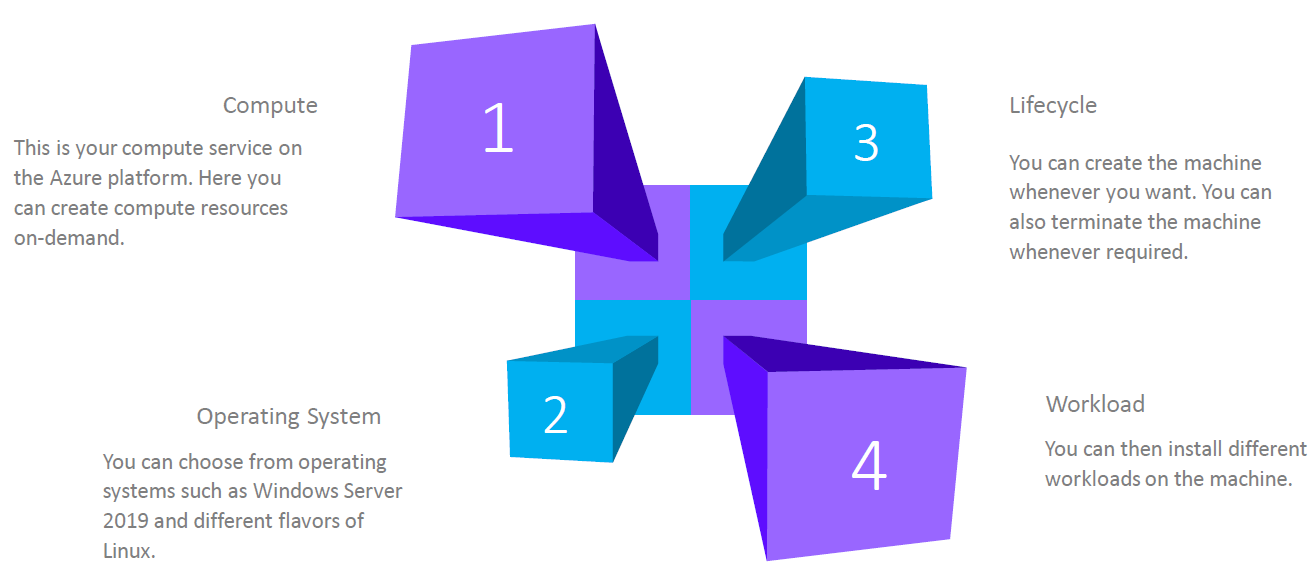
* Try to create the VM where majority of the user resides – This will reduce the latency in the response, when user try access the application.
* The another aspect – we need to consider the cost and availability before selecting the region for the service.
* Usually – when we have used across the globe – we make use of Azure CDN service.

# VIRTUAL MACHINE

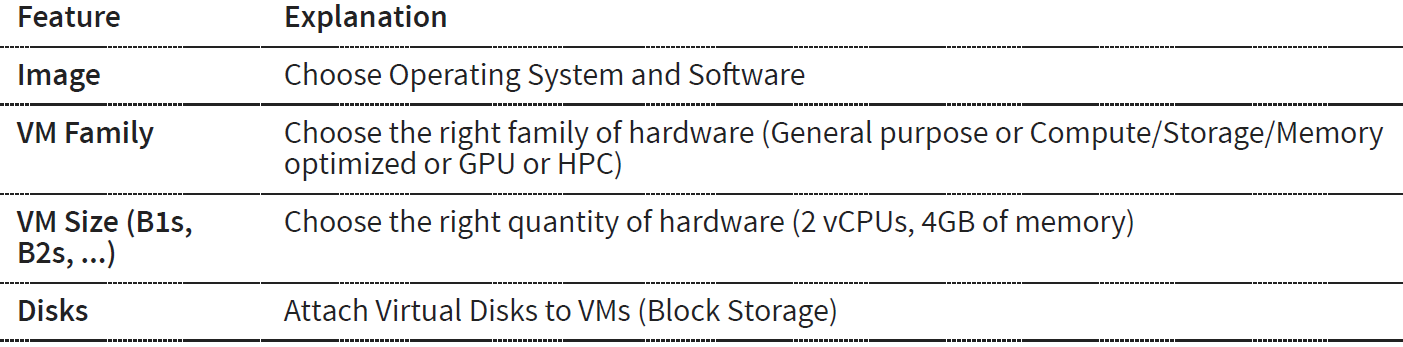
* It is a computer file typically called as an image which behaves like an actual computer.
* It runs in windows, Linux etc. This gives you a flexibility that can run multiple machines in a physical computer.
* Each system can have a different operating system.
* Each of these virtual machines provides its own virtual hardware which includes CPUs, memory, hard drives, network interfaces and other such devices.

## AZURE’S 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)

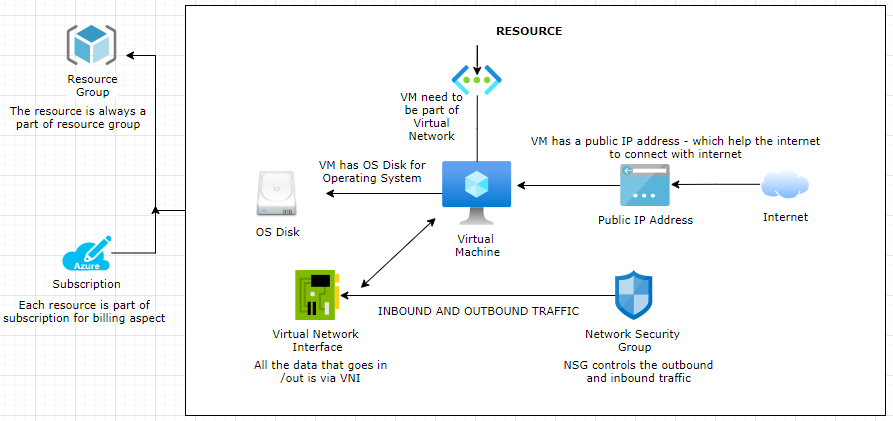


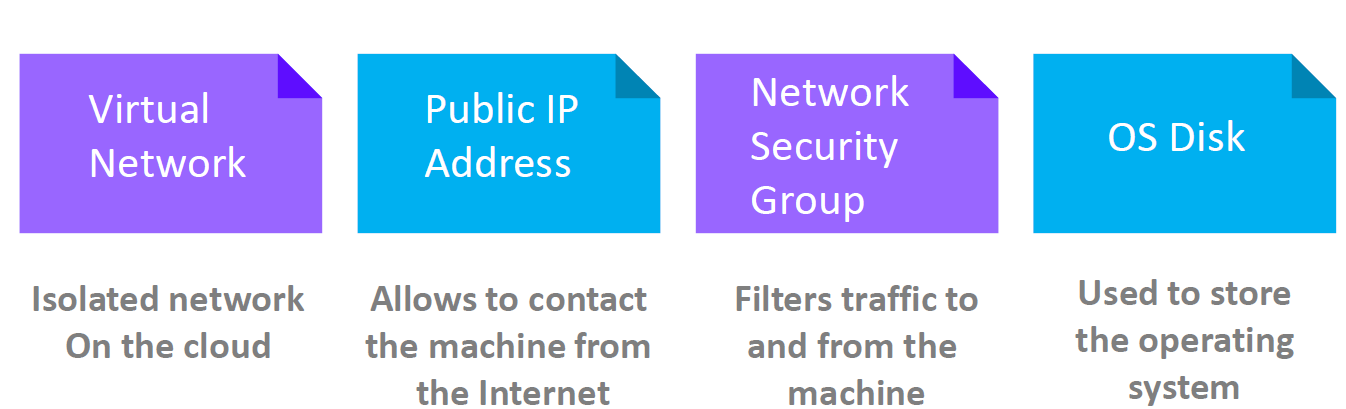
### AZURE VIRTUAL MACHINE – KEY CONCEPTS



## DEPLOYING A VIRTUAL MACHINE

* When we deploy a virtual machine – there are other aspects also get deployed with it.
* VM is a compute resource in Azure Platform. It is an Infrastructure as a service in azure platform





#### 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

#### NETWORK SECURITY GROUP

* It controls all the inbound and outbound traffic

#### VIRTUAL NETWORK

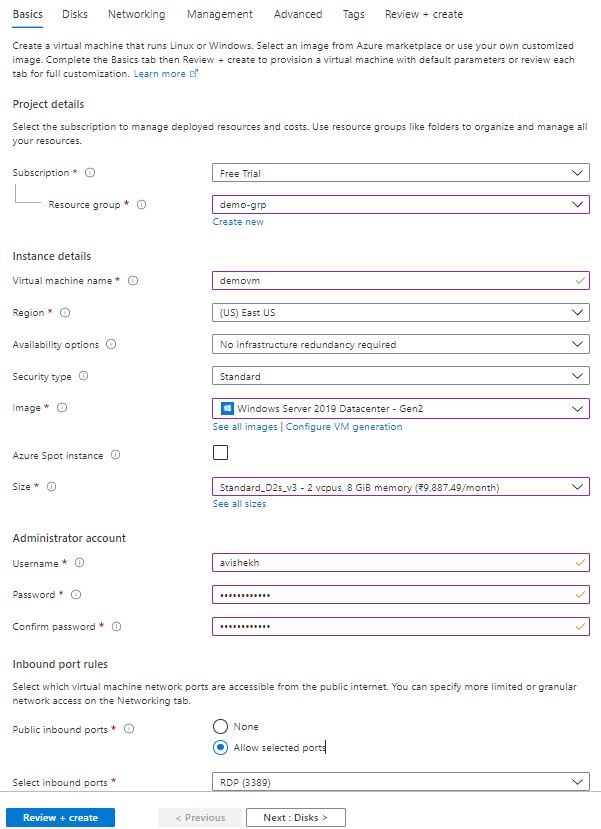
* Every VM is part of a Virtual Network.

#### PUBLIC IP ADDRESS

* The VM are always associated to a public ip address –through which the internet can connect with the VM.

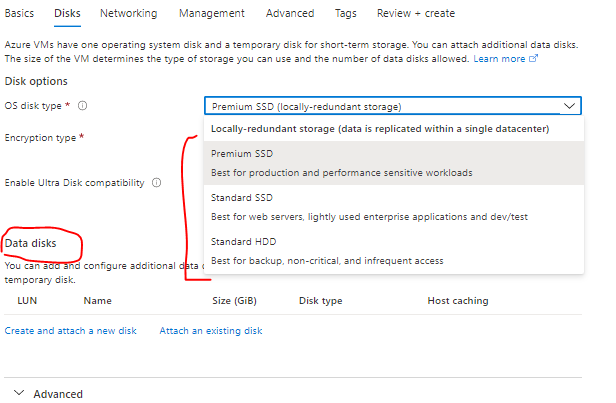
### DEPLOYING WINDOWS VM

#### BASIC



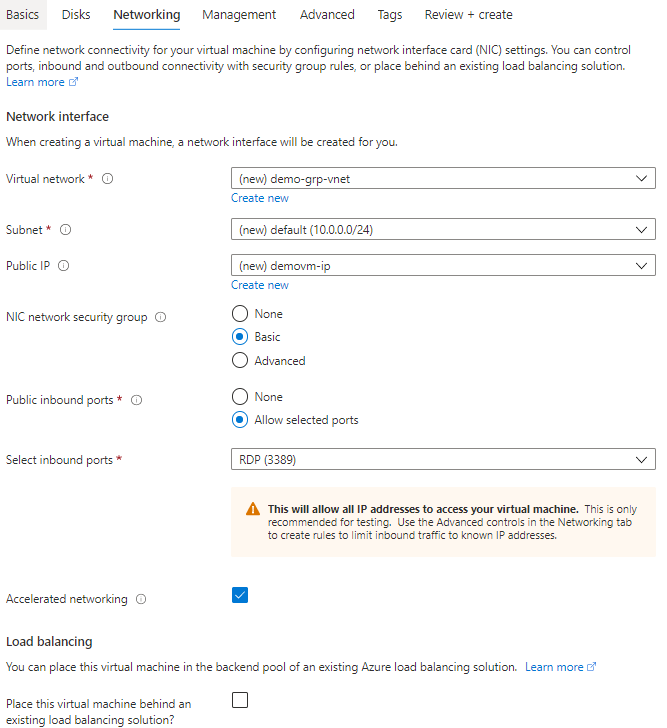
|  |  |
| --- | --- |
| RESOURCE GROUP | The resource group of the VM |
| VM NAME | Name of the Virtual machine |
| REGION | Region in which this VM will be created |
| IMAGE | This is the name of the image – which will be used to set-up OS in the VM. |
| SIZE | This defines the size of the resources in the VM like CPUs and RAM |
| SELECT INBOUND PORTS | This defines on which port the inbound traffic to VM is allowed. For window we user RDP(Remote Desktop) at port 3389 |

#### DISK



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

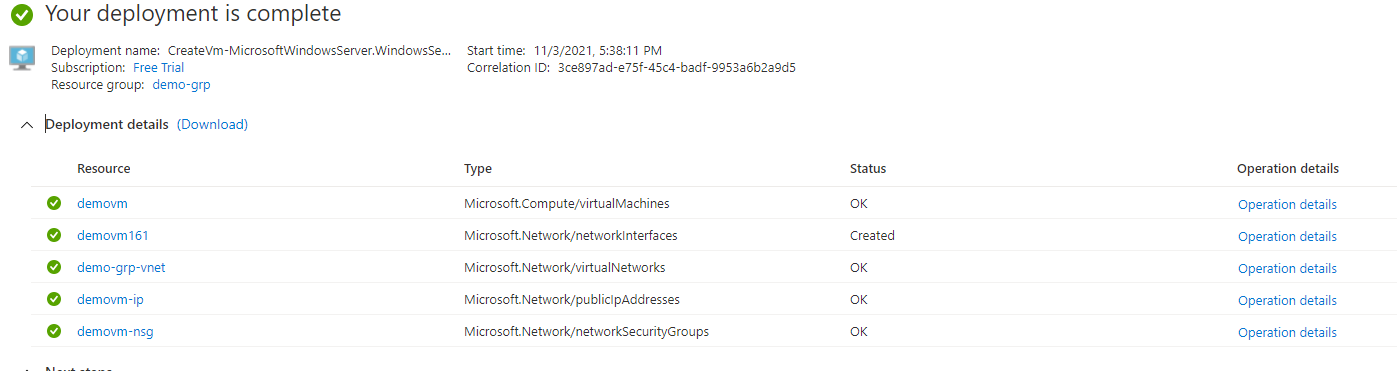
#### NETWORKING



|  |  |
| --- | --- |
| VIRTUAL NETWORK |  |
| SUBNET |  |
| PUBLIC IP ADDRESS | The VM can be accessed on Internet using its public IP address |

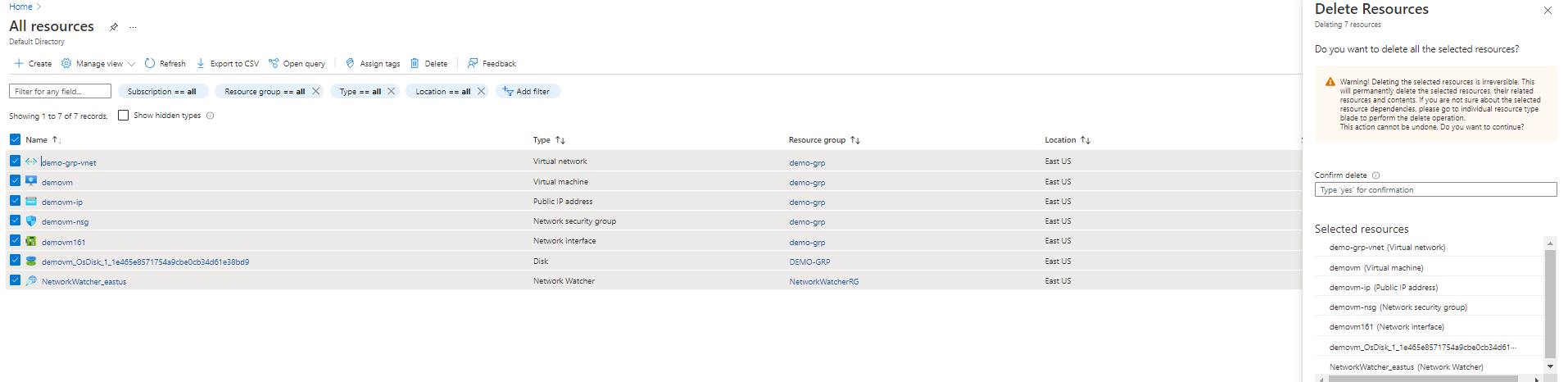
#### VM CREATED

* 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.



#### DELETING A VM

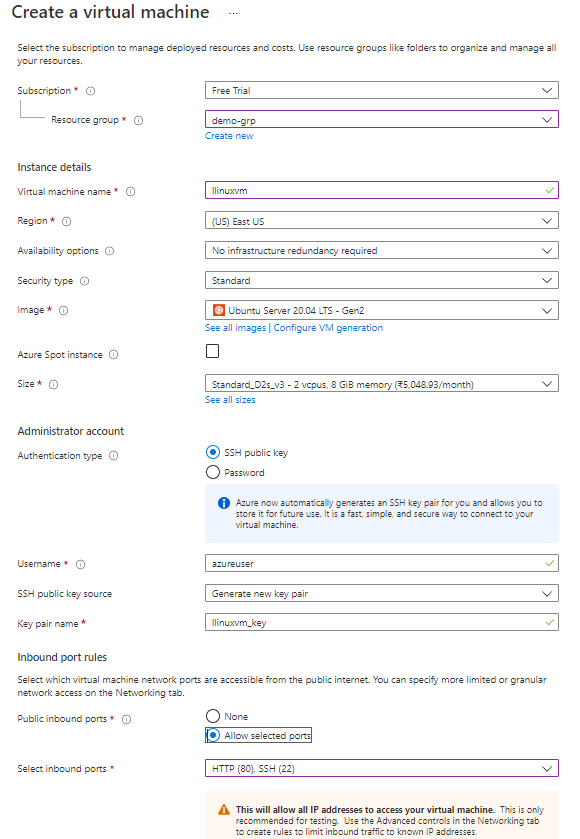
* From the *menu 🡪 All Resources*. Select all the resources which are related to the VM 🡪 Type yes and Delete



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

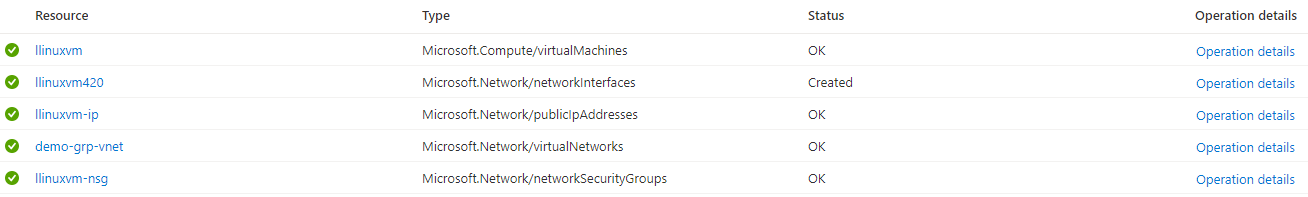
|  |  |
| --- | --- |
|  | * To connect with Windows based VM we use RDP * SSH is used to connect to Linux based VM |
|  | * To connect to the windows VM- download the RDP file * Open the downloaded RDP and Enter the username/password (used while creating the VM) |

### DEPLOYING LINUX VM



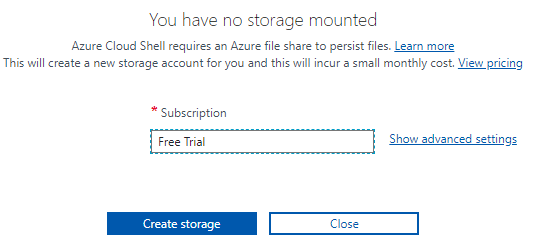
|  |  |
| --- | --- |
|  | * This will show a pop- up to download the SSH key * Download the SSH key – This key will be used to login to VM using SSH. * Note to connect with Linux VM we use SSH. |

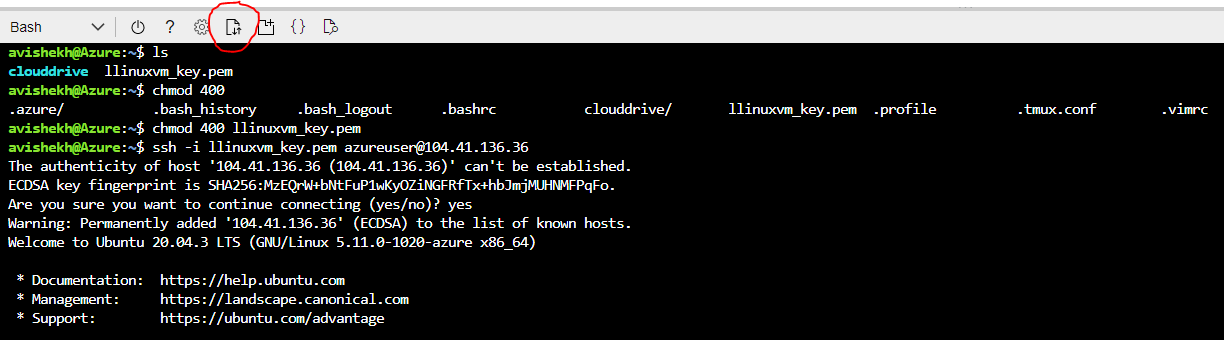
**RESOURCES CREATED WITH VM**



#### CONNECTING TO VM (SSH)

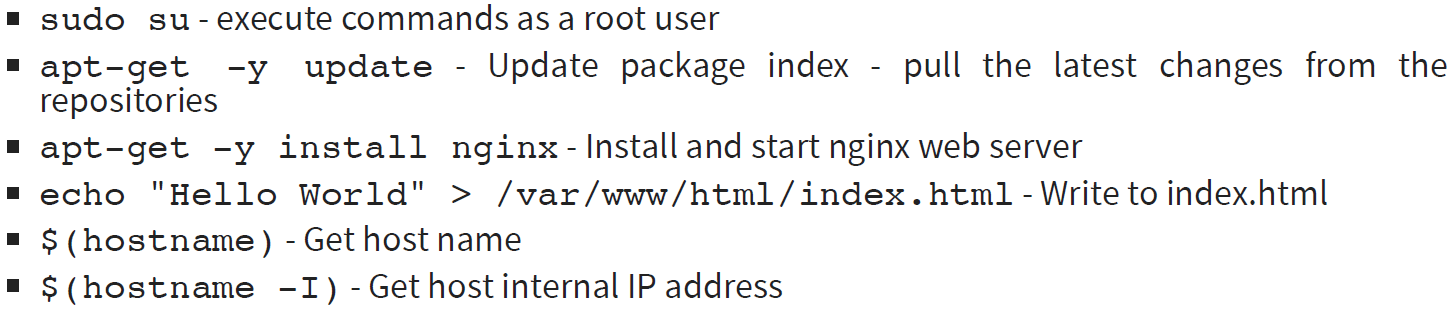
* Open the Cloud Shell and choose Bash
* Select the subscription 🡪 Create Storage



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

### INSTALLING SOFTWARES IN VM

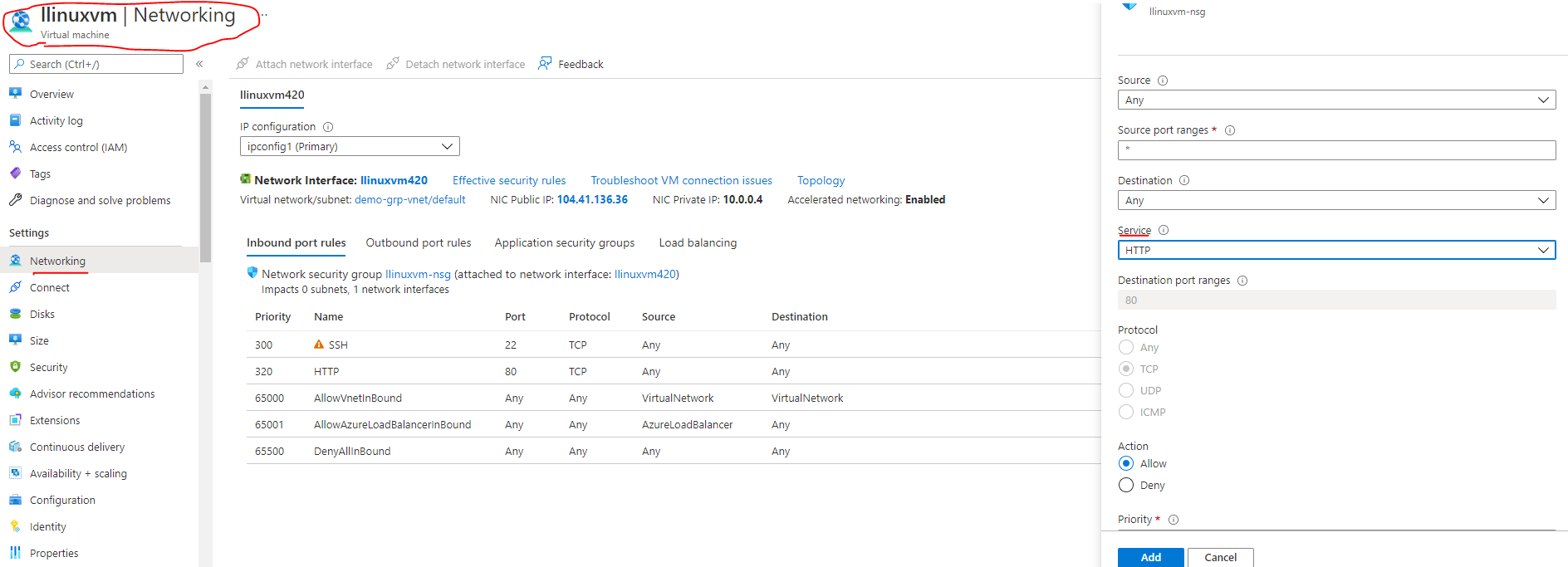
#### INSTALLING NGINX (HTTP SERVER)



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

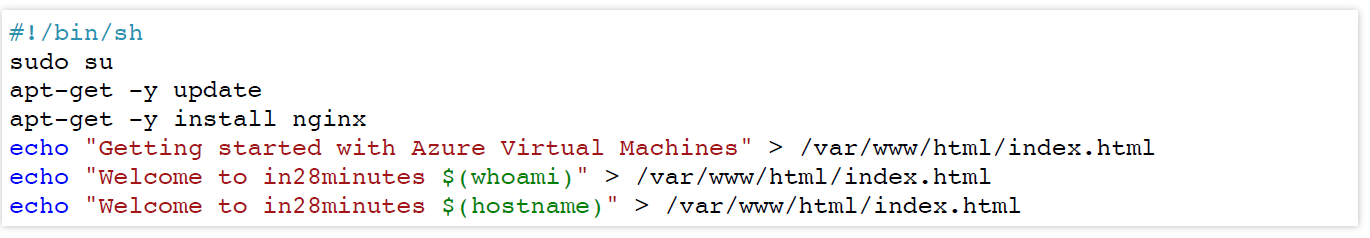
##### ADDING INBOUND PORT RULE

* 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***)



#### 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.



## QUOTA OF VM

* 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

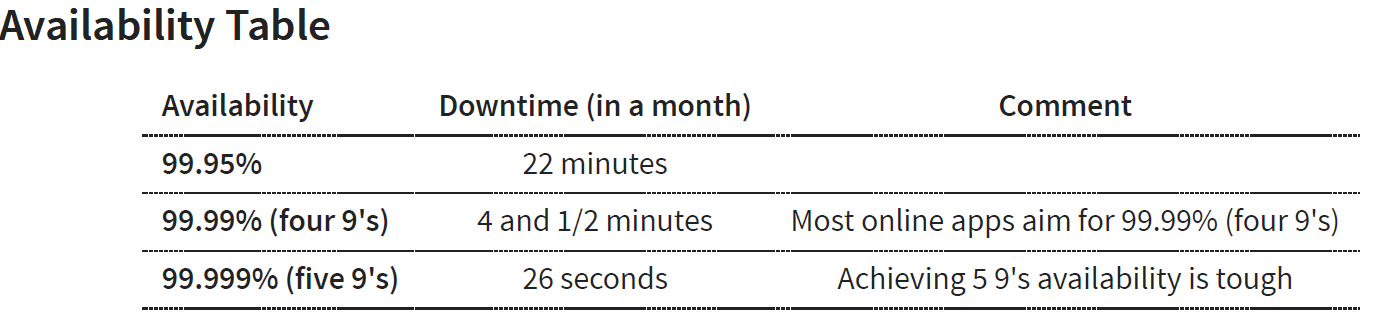


* Note – when we delete the resource group – this will delete all the resources in that resource group.

## AZURE MARKET PLACE

|  |  |
| --- | --- |
|  | * Azure market place offers multiple azure services as a template * For example – if we need a wordpress solution and want to install the solution to one of the VM- we can use the market place. * As wordpress is entirely different company which gives a service via Azure markeplace |

## AVAILABILITY



### 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)

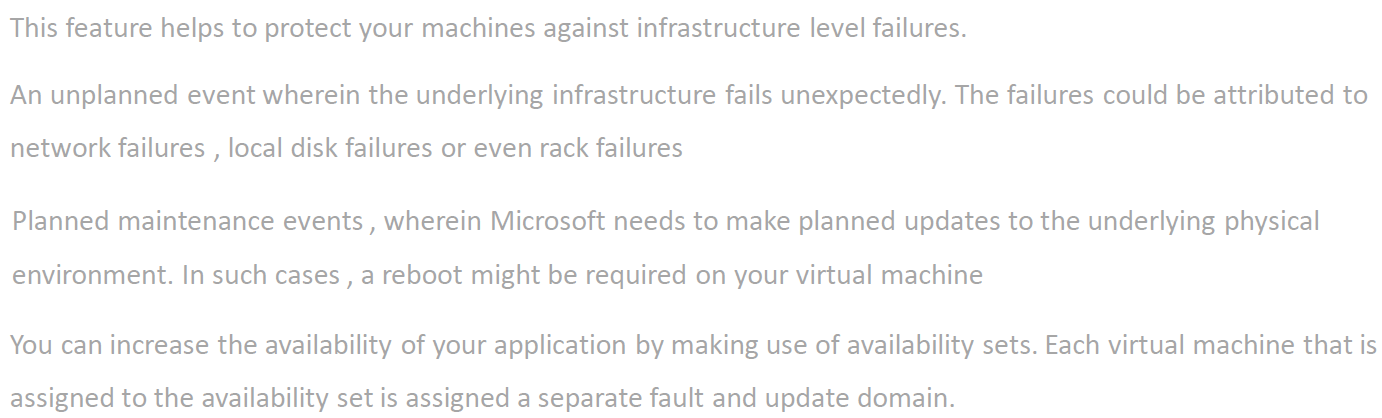
|  |  |
| --- | --- |
| **DISK TYPE** | **AVAILABILITY** |
| Premium SSD or Ultra Disk | 99.9% |
| Standard SSD Managed Disks | 99.5% |
| Standard HDD Managed Disks | *95%* |

Two or more instances in same Availability Set: 99.95%

* Availability set is a logical grouping of VMs
  + **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
* Two or more instances in two or more Availability Zones in the same Azure region: 99.99%
* ***Summary: Create multiple instances in multiple AZs if you want high availability***

### AVAILABILITY SETS

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

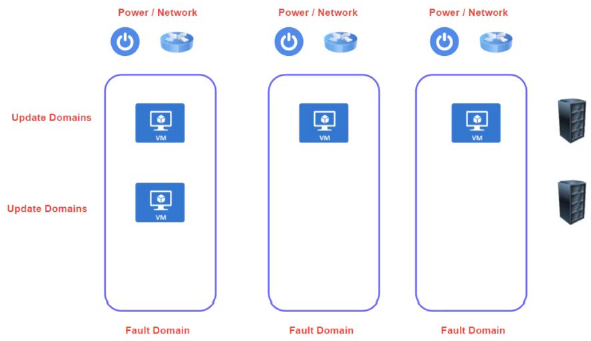


|  |  |
| --- | --- |
|  | * When we deploy the application is multiple VMs – these VM are actually created in phycial server in Azure Data Center. * The phycial server has its dedicated power source and networking. * When we spin-up a VM – we cannot control in which phycial server the VMs are created. * Just in case the physical server goes down (may be due the power source) , then both VMs will go down and hence the application. * To solve this option Azure platform has offers “Availability Set” |

* When a VM is created it is configured to be part of Fault Domain and Update Domain.
* This feature helps to protect your machines against infrastructure level failures.
* An unplanned event wherein the underlying infrastructure fails unexpectedly. The failures could be attributed to network failures , local disk failures or even rack failures
* Planned maintenance events , wherein Microsoft needs to make planned updates to the underlying physical environment. In such cases , a reboot might be required on your virtual machine
* You can increase the availability of your application by making use of availability sets. Each virtual machine that is assigned to the availability set is assigned a separate fault and update domain.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **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 |

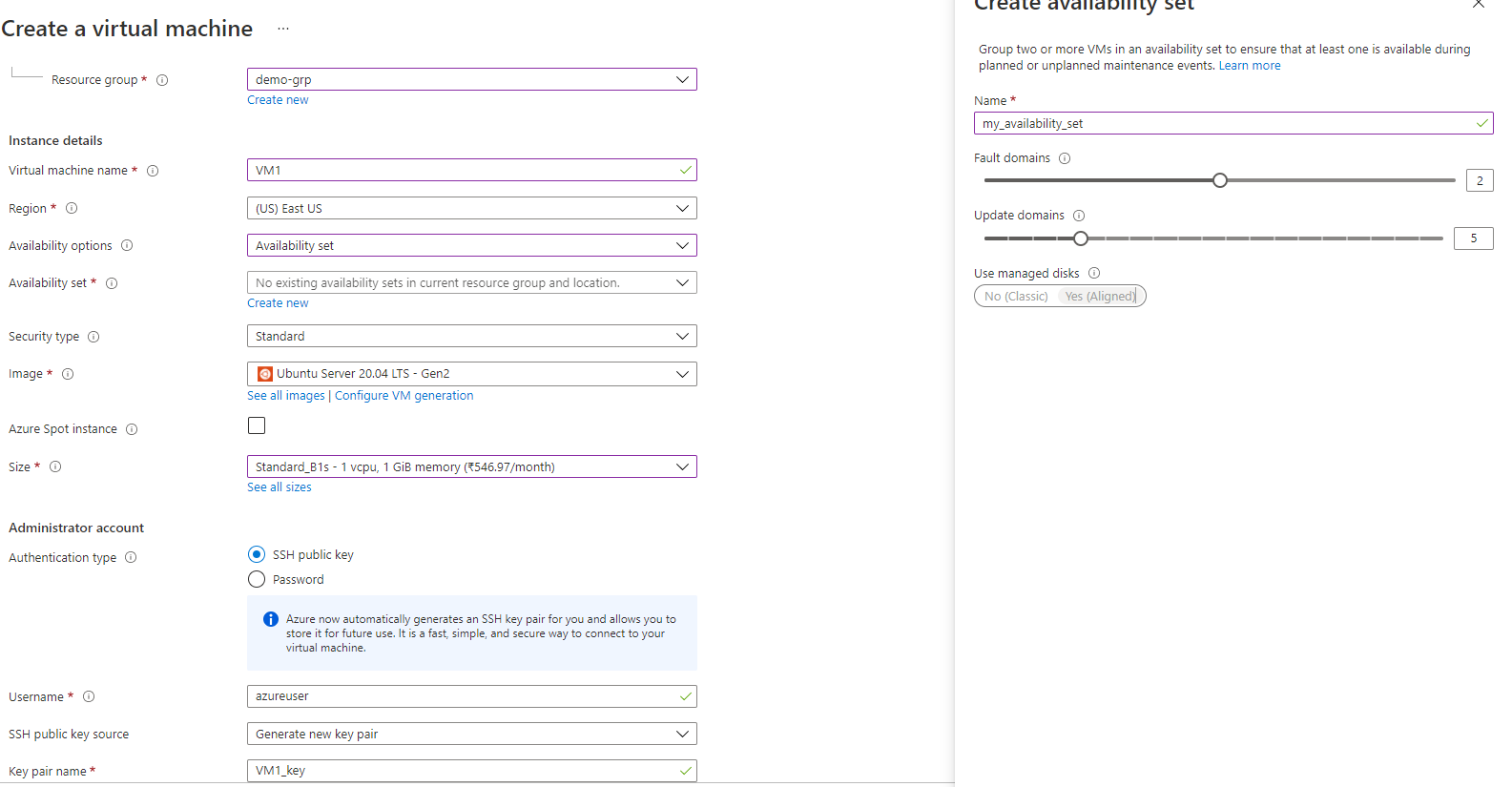
* As the fault domain shares the common power source and network switch – Just in case some goes wrong with this fault domain – then the application will be still available via VM3 and VM4 and vice versa.
* When the physical server needs an update – It will be updated based on update domain. Hence – if update domain UD1 is getting updated then applicatiomn will be available via VM2 and VM4.



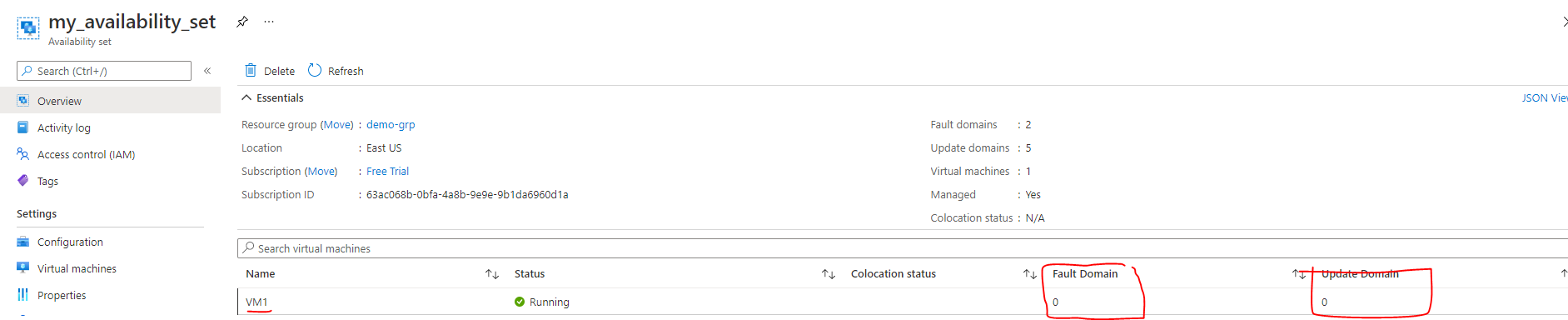
#### EXAMPLE

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

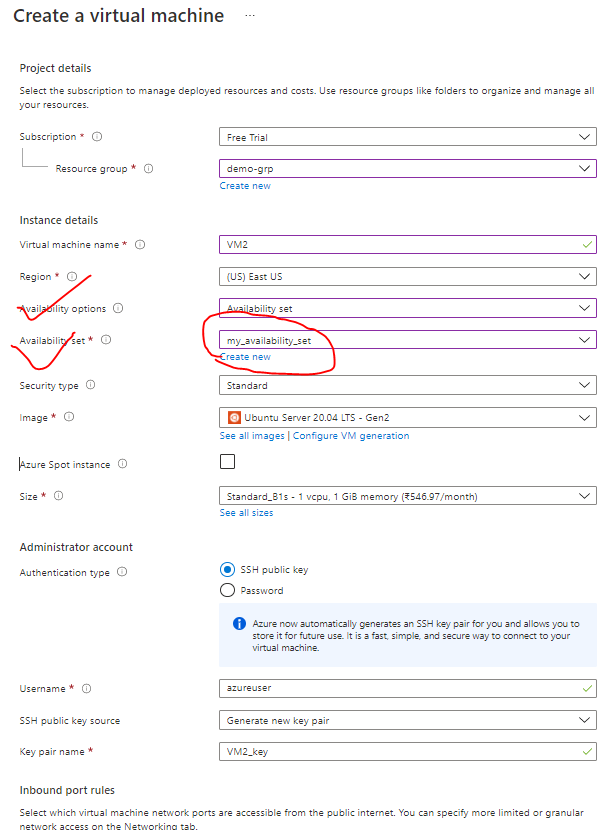
*CREATING THE FIRST VM*



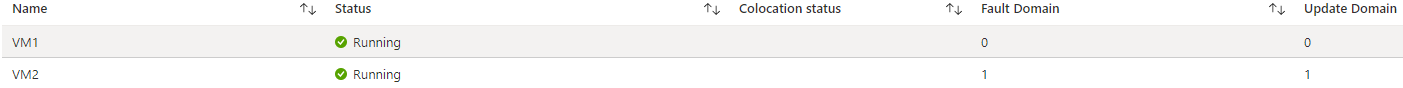
*AVAILABILITY SET STATUS:* The first VM is created in Fault Domain = 0 and update Domain= 0



*CREATING THE SECOND VM IN THE SAME AVAILABILITY SET*



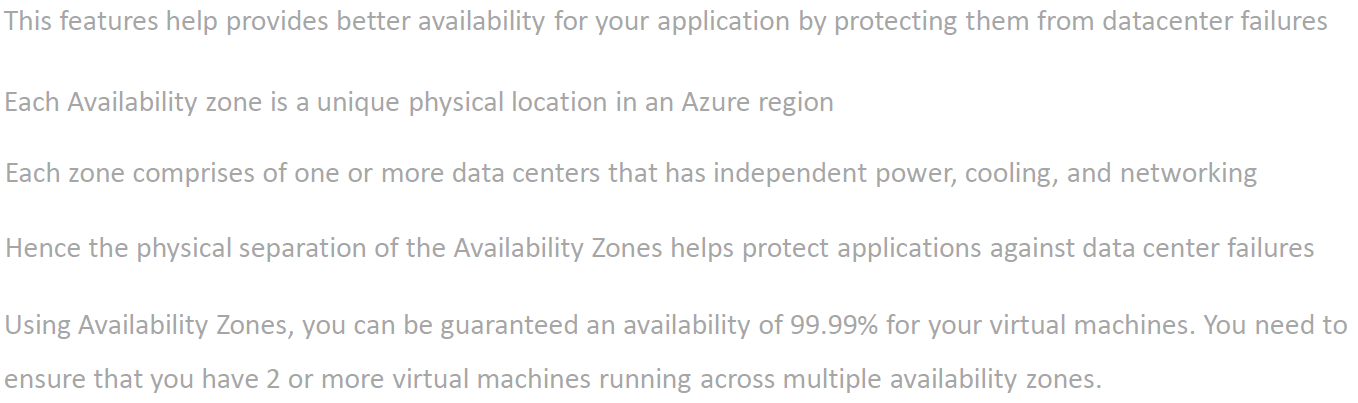
*AVAILABILITY SET STATUS:* The second VM is created in Fault Domain = 1 and update Domain= 1



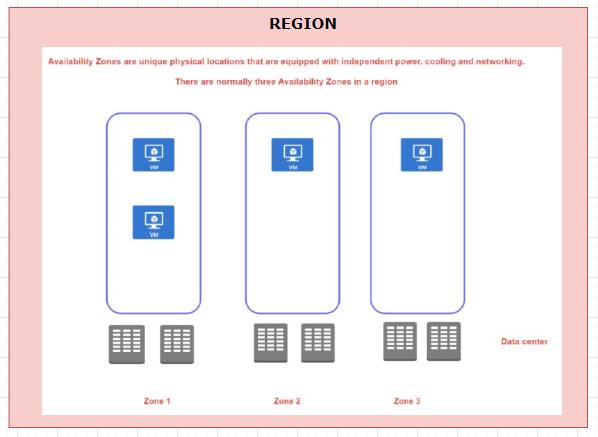
* Hence Azure makes sure that VMs are evenly distributed between the Fault and update domain in a availability zone.
* The configuration of availability set for VM can only be done while creating the VM. VM belong to another availability set cannot eb migrated to. another availability set

**Note: The application deployed to a VM needs to be synced to another VM manually even it belong to same 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 provides better availability for your application by protecting them from datacenterfailures
* Each Availability zone is a unique physical location in an Azure region
* Each zone comprises of one or more data centers that has independent power, cooling, and networking
* Using Availability Zones, you can be guaranteed an availability of 99.99% for your virtual machines. You need to ensure that you 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.

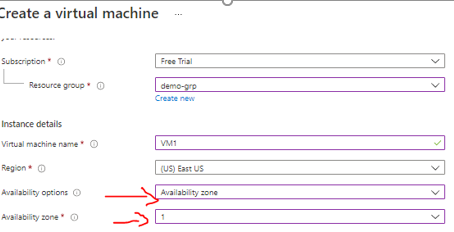
### NOTES ON AVAILABILITY ZONES AND SET

* There is no extra cost involved in creating availability zone or sets. But the 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 (ZONE -0 ; REGION – EAST US)



**VMS IN DIFFERENT AVAILIBILITY ZONE**

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

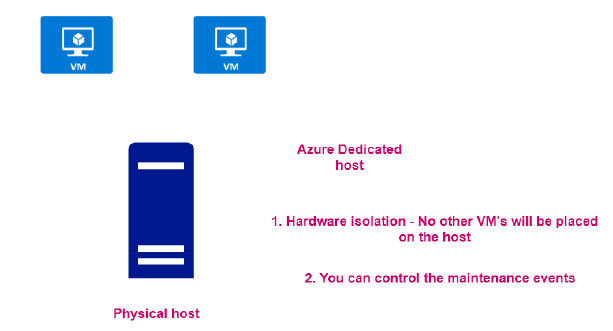
**WHY SHOULD I CREATE AVAILABILITY SETS IF WE HAVE AVAILABILITY ZONE?**

* 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

## VIRTUAL MACHINE SCALE SETS

* Virtual machine scale set simplify the creation and management of multiple VMs(group of VMs)
* VM scale set also allow as to add a load balancer
* Supports manual and auto scaling
* Distribute VM instances across multiple Availability Zones
* Supports 1000 VM instances in a single scale set.

## AZURE DEDICATED HOST



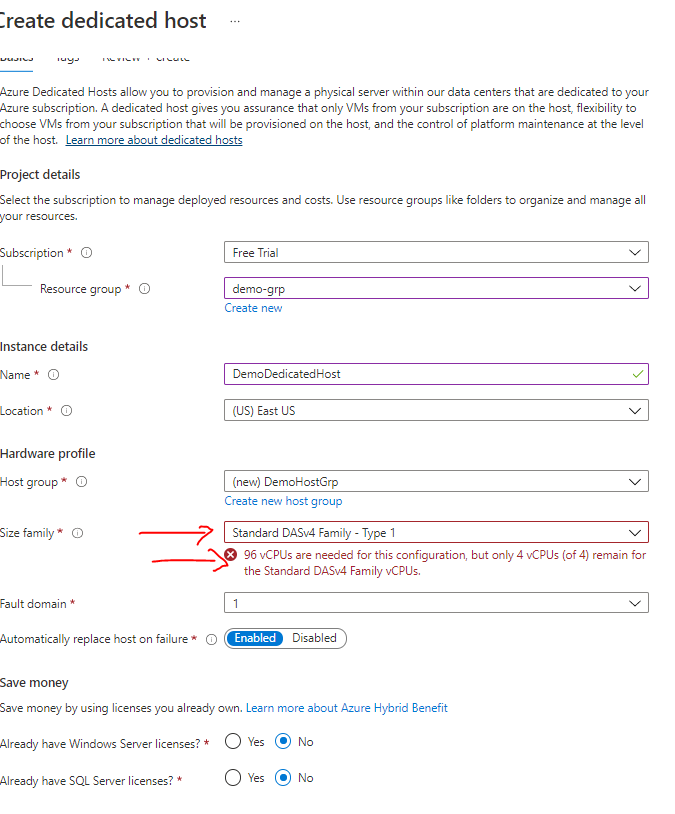
* When we create VMs, they are get created in physical server/ host in the azure data center
* In Azure – it come with a capability to assign the entire physical host as a resource. Note this capability are usually leveraged by large scale organization.

#### 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.

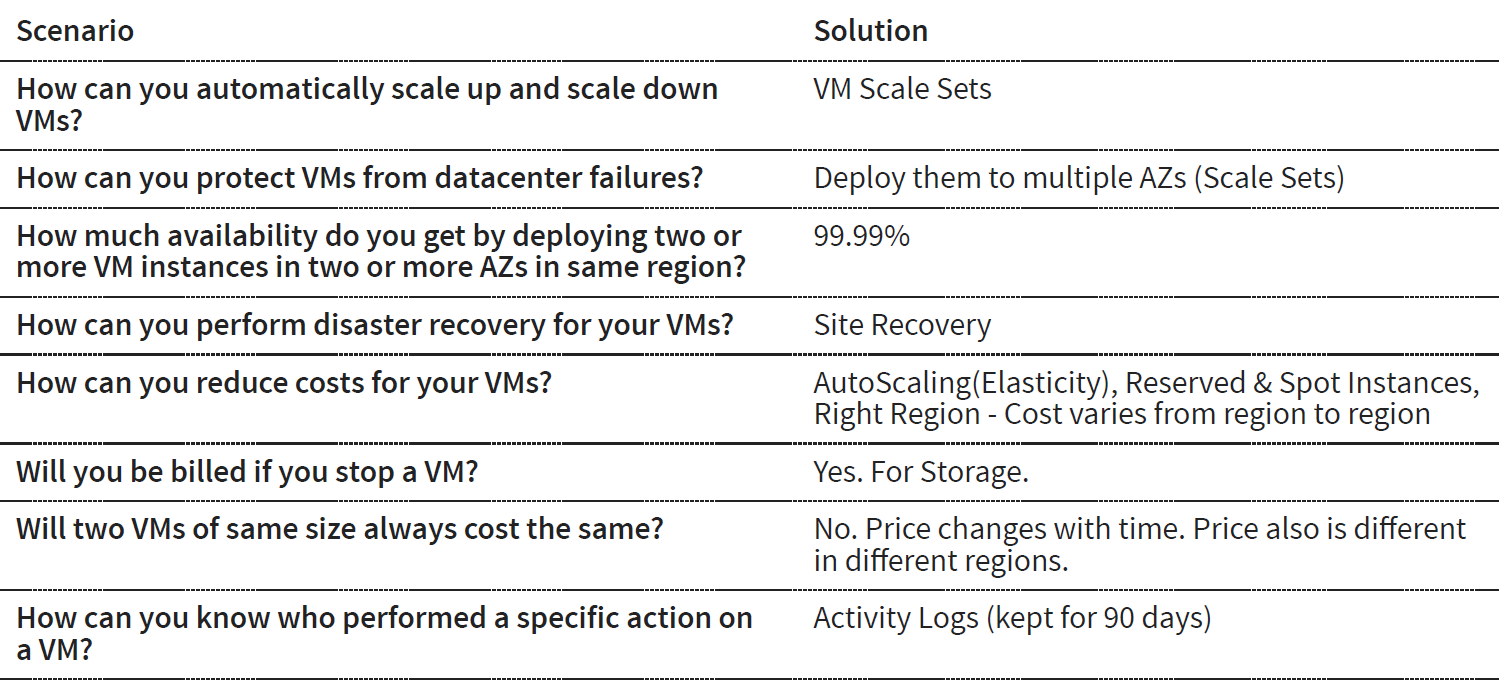


## AZURE SPOT INSTANCES

## WORKLOAD

|  |  |
| --- | --- |
|  | * ***Workload is unit functionality which can be an application or service.*** * For example – If we have a web application which we want to host or a database server they are called as “workload”. |

## IMPORTANT USE CASES AND SOLUTIONS



# AZURE VIRTUAL NETWORK

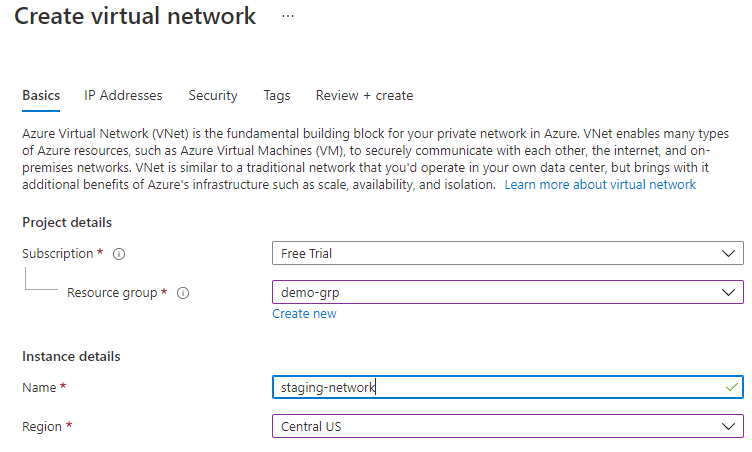
* The Virtual network is a is always a range of IP address.
* When we create a VM it is always get a private IP address. If the private IP of the VM fall in the range of virtual machine, then the machine will be part that virtual network. Hence, The private IP helps in locating the VM in the vitual network.
* The Azure Virtual Network service is used to define an isolated network in Azure. The virtual network can then be used to host the resources such as Azure virtual machines.
* The Azure virtual network gets assigned an address space (IP address range) which we can specify when we create an Azure virtual network.

## CREATING A VIRTUAL NETWORK

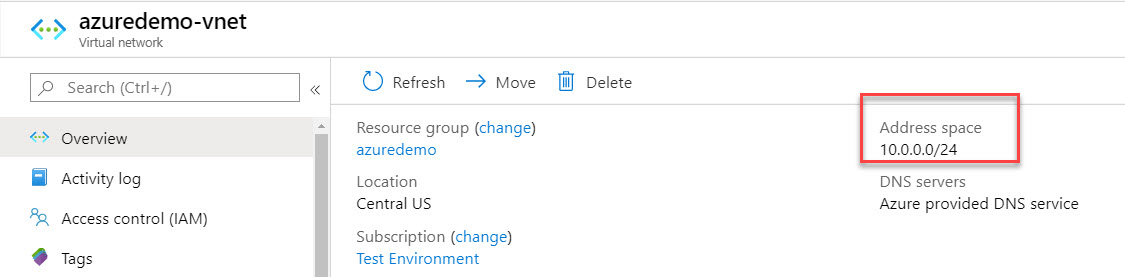
* Usually the Virtual network is created aling with the VM. But we can also able to create Virtual machine upfront and the tie the VM into it.



* ***Step 1***: Lets create a Virtual al network. This wizard will create subnet as well in the SUBNET as well network.



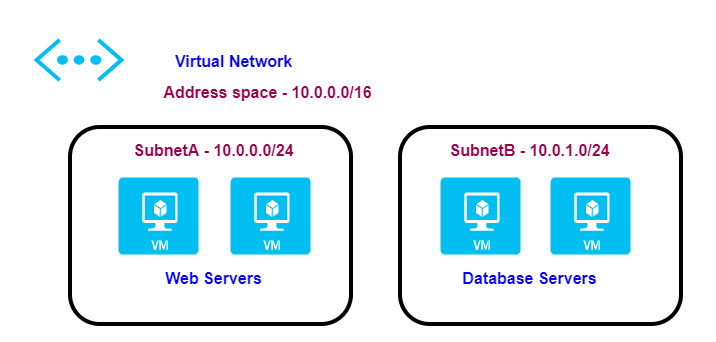
* *Note: When we create a VM needs to be part of virtual network*
* Virtual network has something called IP address range / address space as shown below

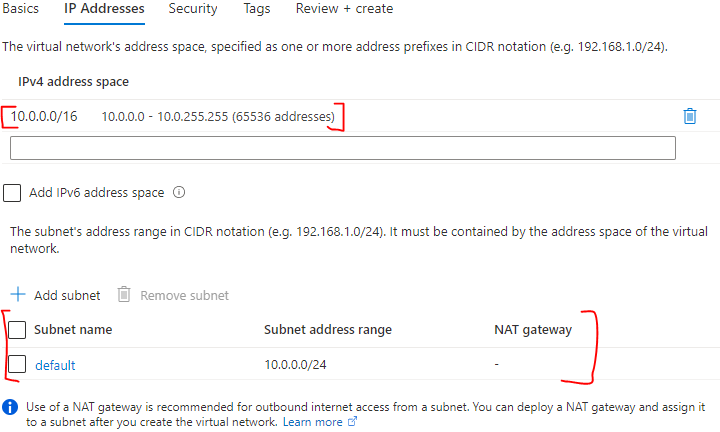


## CREATING A SUBNET IN VIRTUAL NETWORK

After creating a Virtual Network – We can then add subnets to the Azure virtual network. This helps divide the network into more logical segments.

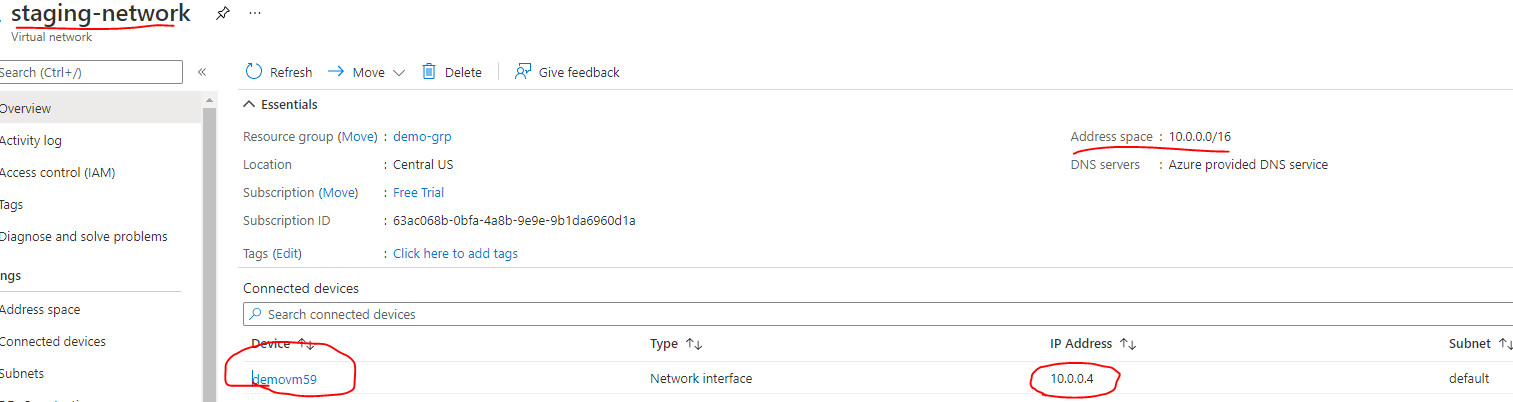
An example is shown below of having multiple subnets. You could have one subnet named SubnetA in the virtual network to host your Web servers and another subnet to host the Database servers.





## CREATING A VIRTUAL MACHINE IN VIRTUAL NETWORK

* While creating a VM we need in a virtual network – we need to select the same region as of the Virtual Network.
* If the correct region is selected in the “Basic” tab – then only the Virtual network will be visible in “Networking” tab.



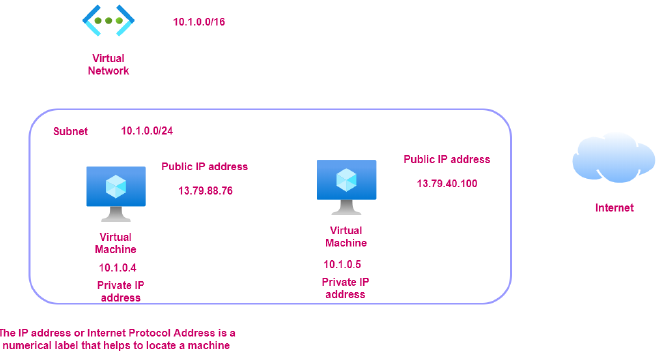
* The above diagram shows the VM created in the virtual network.
* The IP of the VM lies within the range of Virtual Network

*Questions*

1. *If I have pre-created VM (which belong to some another virtual network) can I change the Virtual network of the VM ? - No*
2. *Can a VM can be a part of two different Virtual Network? – No*

## COMMUNICATION ACROSS VIRTUAL MACHINES IN A VIRTUAL NETWORK

* The virtual machines within the Virtual network can communicate using their private IP address.

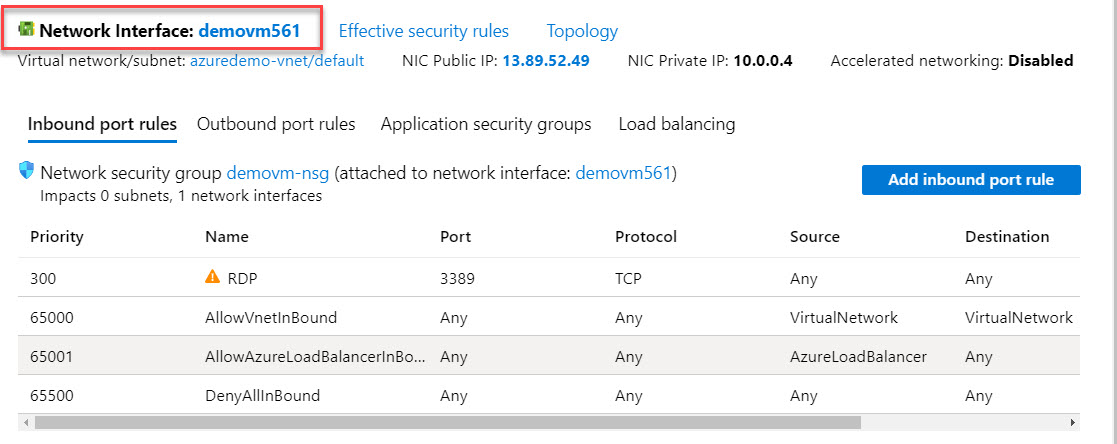


## NETWORK SECURITY GROUP

* Network security group is used to filter the in-bound and outbound traffic which is flowing to the VM.
* All the data flows into VM go through the Virtual Network Interface. Hence – when we access the VM using the public IP address. The traffic flow via Virtual Network interface.
* The Network security group (which is attached to the Virtual Network Interface) - has set of rules which controls / filters the inbound and outbound traffic. It’s like a basic firewall.
* The NSG rules can also be applied on subnet layer as well.
* All the inbound and outbound rules can be set from “Networking” in VM dashboard.



* By default, all traffic into a virtual machine is DENIED. We must explicitly add rules to allow traffic into a virtual machine
* There are also outbound rules to control the traffic flowing out of the virtual machine. By default, all traffic outbound onto the Internet is *allowed*.
* The inbound / outbound rules include the rules around protocol, port number, Source and Destination and Priority.
* The Network security rule are evaluated from top to bottom. If the rule matches in rule, rest of the rules will be ignored



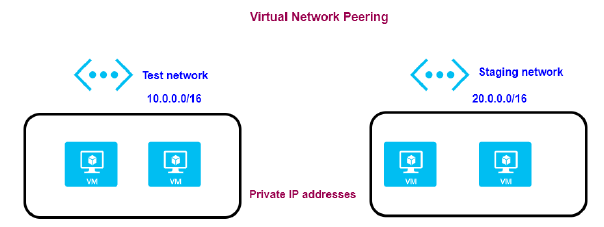
## APPLICATION SECURITY GROUP

|  |  |
| --- | --- |
|  | * Let’s say – we have VM which has webserver which is in turn communicating with a VM having DB server. * Application Security Group is one of the ways of filtering traffic using IP address. * In the above example if we have multiple VM(webserver) communicating with the DB server. We create an Application Security Group – which has list of all such VMs. * At the DB VM side – we have to make sure that its NSG should accept the incoming request from the Application Security Group- which in turn means DB VM is allowing the incoming request from webservers |

## NETWORK CONNECTIVITY OPTIONS

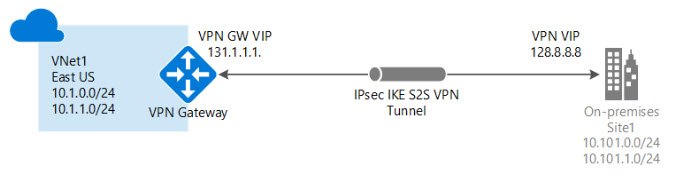
### **VIRTUAL NETWORK PEERING**

* Virtual Network Peering is used to connect two Azure virtual networks together via the backbone network.
* Azure supports connecting two virtual networks located in the same region or networks located across regions.
* Once we enable virtual network peering between two virtual networks, the virtual machines can then communicate via their private IP addresses across the peering connection.
* We can also peer virtual networks that are located across different subscriptions.
* The virtual networks can't have overlapping CIDR blocks.



### **SITE-TO-SITE VPN CONNECTION**

* A Site-to-Site VPN connection is used to establish a secure connection between an on-premise network and an Azure network via the Internet.



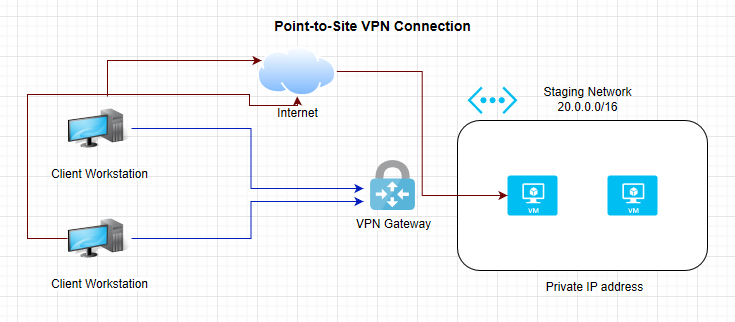
* On the on-premise side, you need to have a VPN device that can route traffic via the Internet onto the VPN gateway in Azure. The VPN device can be a hardware device like a Cisco router or a software device ( e.g Windows Server 2016 running Routing and Remote services). The VPN device needs to have a publically routable IP address.
* The subnets in your on-premise network must not overlap with the subnets in your Azure virtual network
* The Site-to-Site VPN connection uses an IPSec tunnel to encrypt the traffic.
* The VPN gateway resource you create in Azure is used to route encrypted traffic between your on-premise data center and your Azure virtual network.

### **POINT-TO-SITE VPN CONNECTION**

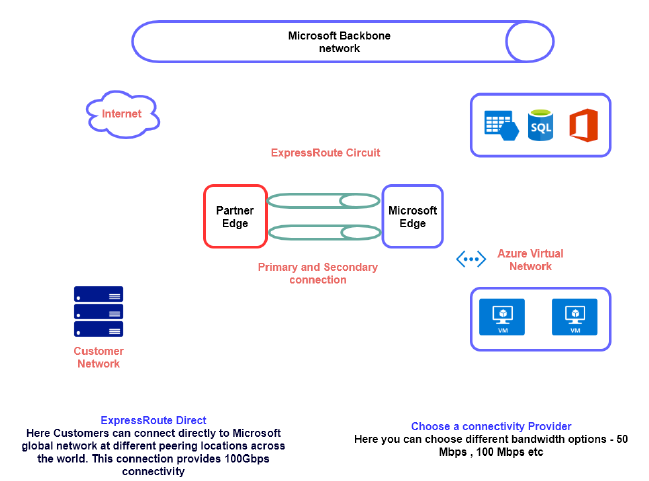
* A Point-to-Site VPN connection is used to establish a secure connection between multiple client machines and an Azure virtual network via the Internet.

#### USE CASE

* If we have multiple Client workstation which needs a connection to the Virtual Network.
* Usually, the VM can be accessed using the public IP address.
* Since the connections is via internet hence for security standpoint – the client workstations should talk to VM via private IP address. For the communication via Private IP address with the VM , the client workstation has to connect via VPN.
* For enable this set-up we need to deploy a resource called VPN gateway resource and configure Point to site VPN connection.
* In addition to that the set up make use of certificates for authentication to make the connection more secure



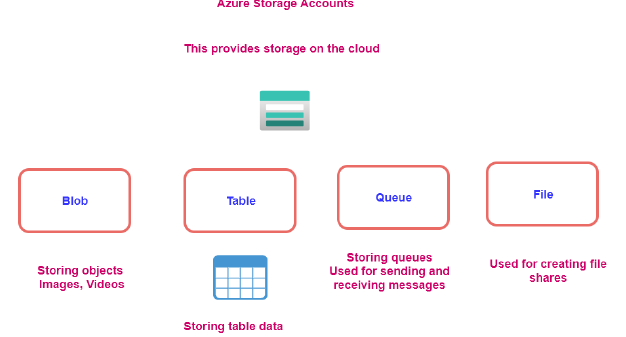
## AZURE EXPRESS ROUTE



# AZURE CORE SERVICES

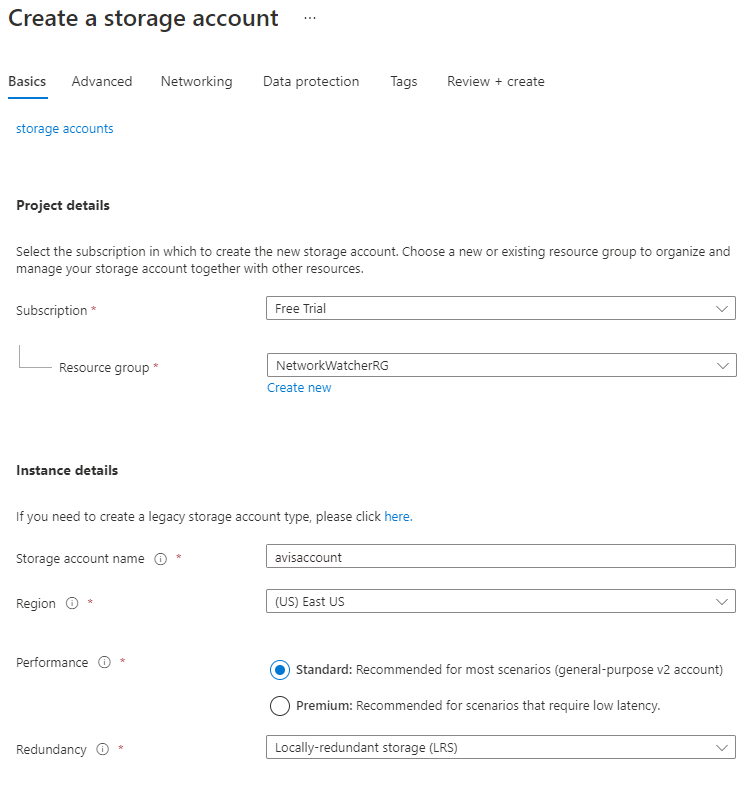
## AZURE STORAGE

* Azure storage account provides the storage on cloud and we use a Azure storage service account for the same.
* Depending upon the service account we can use of following services like Blob (To store objects , Images and Videos),Table(To store table data),Queue(Storing queues which is used for sending and receiving message),File(Used for creating file share).



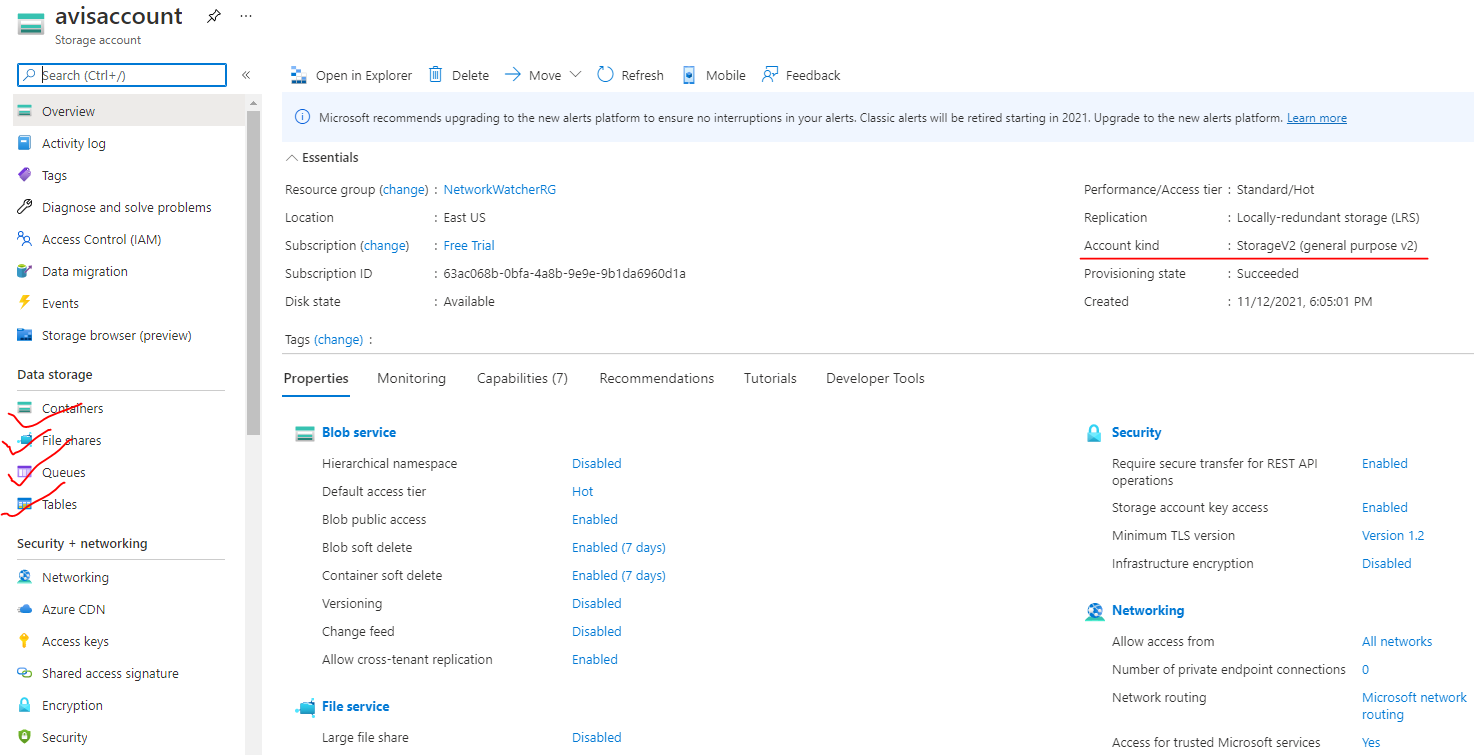
### CREATING A STORAGE ACCOUNT

* Create Resource 🡪 Storage Account
* Give the storage Account name



### STORAGE ACCOUNT

* The general-purpose storage account is equipped with Container(Container for Storing BLOBs like images , video etc.),File(File Share), Queue(messages) and tables



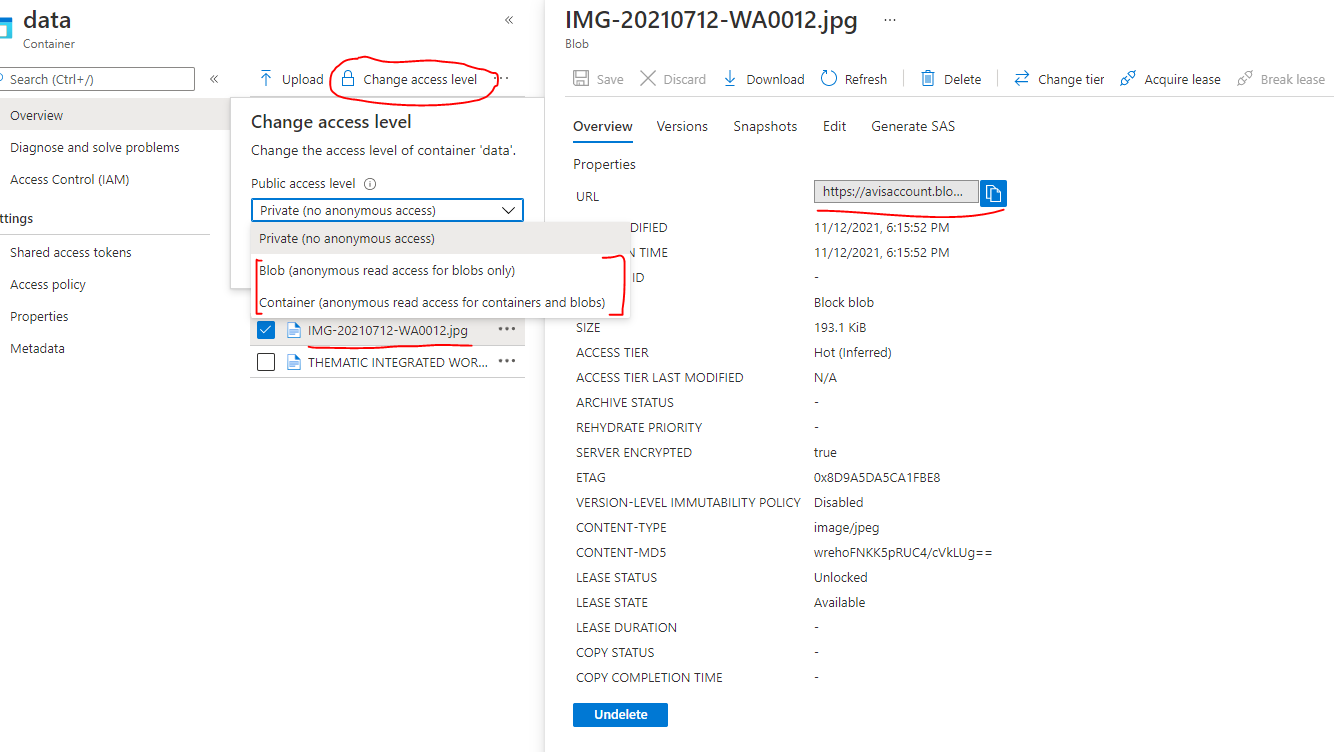
### CONTAINER

#### CREATING A CONTAINER

|  |  |
| --- | --- |
|  |  |
| * Enter the name of the container. * The container will hold all the blobs like image and videos * Using the upload option we can able to upload the images and videos * Note : ***Anything uploaded in a container will have unique url.*** | |

##### ACCESSING THE BLOB

* All the BLOB items has a unique url. Click on the items to view its property
* The URl field will give the get the unique url of the BLOB Item(e.g : <https://avisaccount.blob.core.windows.net/data/IMG-20210712-WA0012.jpg> )
* ***The items will be accessible when the Access level is not private.***



##### ACCESSING TIERS OF BLOBS



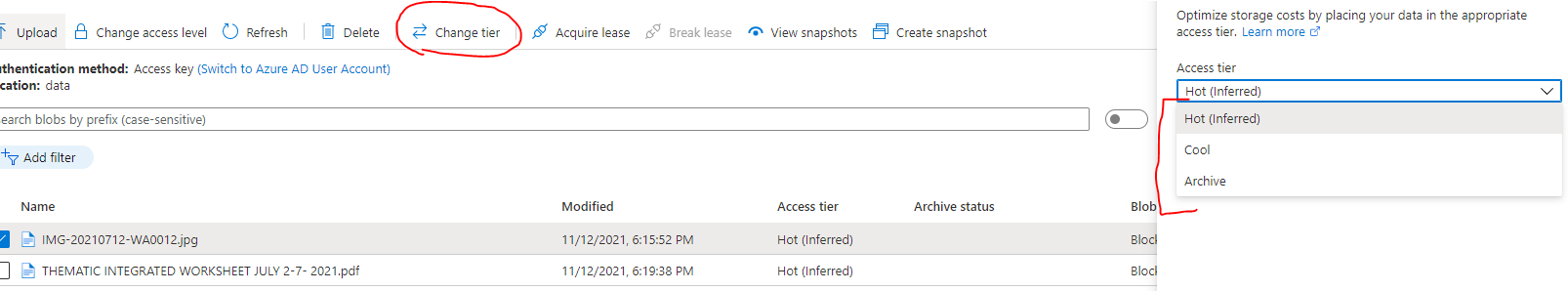
* Data stored in the cloud grows at an exponential pace. To manage costs for the expanding storage needs, it can be helpful to ***organize the data based on how frequently it will be accessed and how long it will be retained***.
* Azure storage offers different access tiers so that we can store the blob data in the most cost-effective manner based on how it is being used.

*Azure Storage access tiers include*

* **HOT TIER** - An online tier optimized for storing data that is accessed or modified frequently. The Hot tier has the highest storage costs, but the lowest access costs.
* **COOL TIER** - An online tier optimized for storing data that is infrequently accessed or modified. Data in the Cool tier should be stored for a minimum of 30 days. The Cool tier has lower storage costs and higher access costs compared to the Hot tier.
* **ARCHIVE TIER** - An offline tier optimized for storing data that is rarely accessed, and that has flexible latency requirements, on the order of hours. Data in the Archive tier should be stored for a minimum of 180 days.

**Note** : Setting the access tier to "Archive" will make your blob inaccessible until it is rehydrated back to "Hot" or "Cool", which may take several hours.

The Access Tiers can be changed as shown below. Select the BLOB item 🡪 Change Tier 🡪 Select the Access Tier.



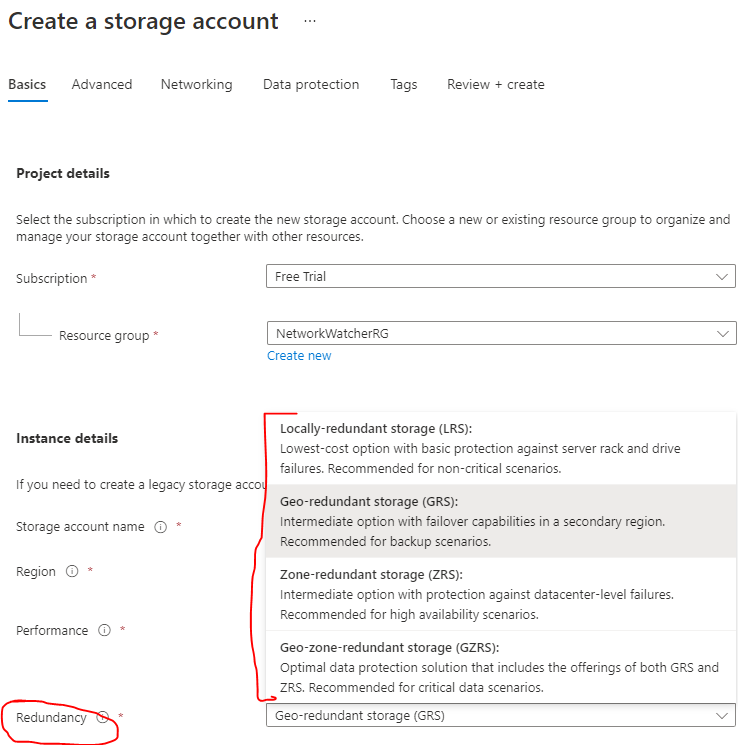
#### DATA REDUNDANCY IN AZURE STORAGE ACCOUNT

##### WHAT IS REDUNDANCY?

|  |  |
| --- | --- |
|  | * When we store a data in a storage account – it is basically get stored in a physical storage device. * When azure store in the data in the storage devices – it stores multiple copies of the data * This helps in protecting the planned and unplanned events , transient hardware failures , network and power outage. |

The different statergies Azure uses to make multiple copies of the data are following

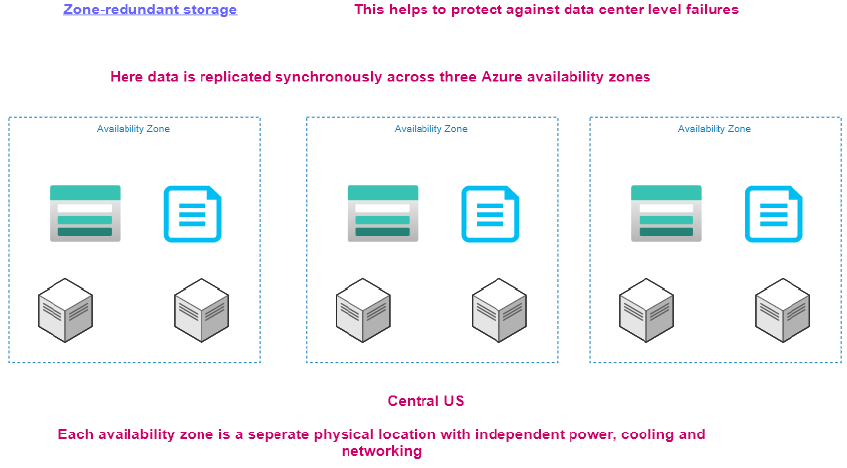
1. **LOCALLY-REDUNDANT STORAGE**
2. **GEO-REDUNDANT STORAGE**
3. **ZONE-REDUNDANT STORAGE**
4. **GEO-ZONE-REDUNDANT STORAGE**



##### LOCALLY-REDUNDANT STORAGE

|  |  |
| --- | --- |
|  | In this strategy of making the copies of data   * In an Azure data center 3 copies of data are made in different storage devices * Hence event if there is failure (like rack or device failure) in one of the storage device – the data will be still available |

##### ZONE-REDUNDANT STORAGE



In Zone Redundant Storage

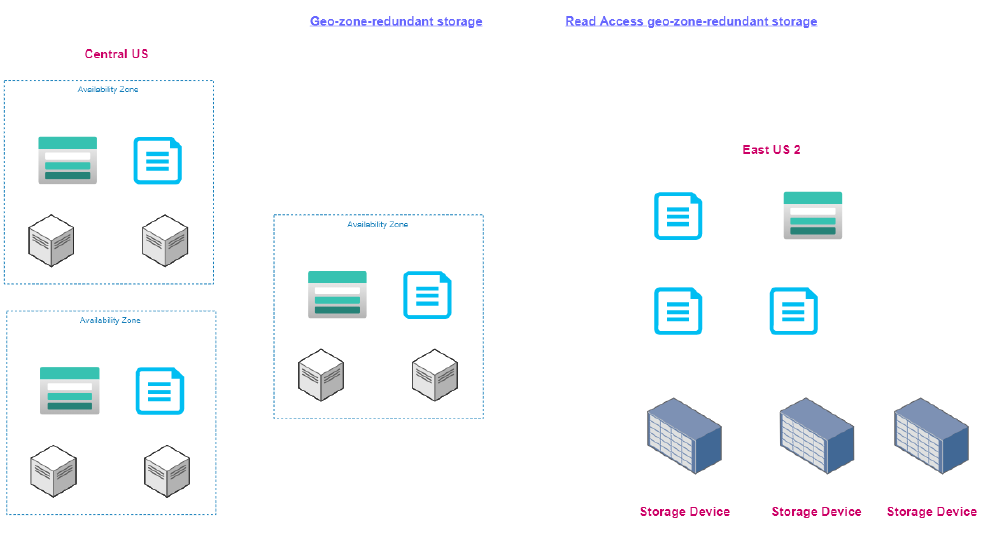
* This help in protecting against failure in a data center.
* Here the data is copied to 3 availability zones – Hence the data will be available even if a data center fails.

Note:

* Each availability zone reside in separate physical location.
* In this strategy- if the entire zone is not available (like Central US) , then all the availability zone will not be available. In this case we can go with Geo Redundant Storage.

##### GEO-ZONE-REDUNDANT STORAGE

* The data is first replicated to availability zones in a particular zone.
* After that the data is replicated to secondary region



### FILE SHARE

### TABLE

* Azure Table storage is a service that stores **non-relational structured data (also known as structured NoSQL data)** in the cloud, providing a key/attribute store with a schemaless design. Because Table storage is schemaless,
  + it's easy to adapt your data as the needs of your application evolve.
  + Access to Table storage data is fast and cost-effective for many types of applications and is typically lower in cost than traditional SQL for similar volumes of data.
* We can use Table storage to store flexible datasets like user data for web applications, address books, device information, or other types of metadata your service requires.
* **We can store any number of entities in a table, and a storage account may contain any number of tables, up to the capacity limit of the storage account**.

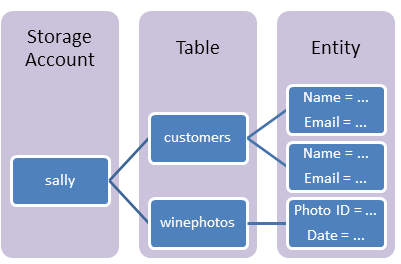
#### WHAT IS TABLE STORAGE

* Azure Table storage stores large amounts of structured data. The service is a NoSQL datastore which accepts authenticated calls from inside and outside the Azure cloud. Azure tables are ideal for storing structured, non-relational data.

*Common uses of Table storage include*

* Storing TBs of structured data capable of serving web scale applications
* Storing datasets that don't require complex joins, foreign keys, or stored procedures and can be denormalized for fast access
* Quickly querying data using a clustered index
* Accessing data using the OData protocol and LINQ queries with WCF Data Service .NET Libraries
* We can use Table storage to store and query huge sets of structured, non-relational data, and the tables will scale as demand increases.

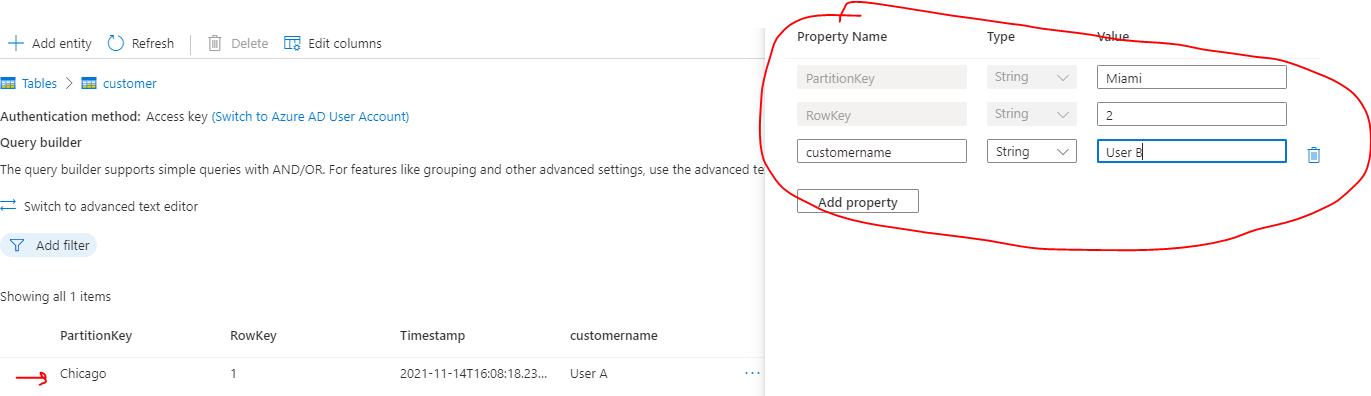
#### TABLE STORAGE COMPONENTS



|  |  |
| --- | --- |
| **URL FORMAT** | * Azure Table Storage accounts format: http://<storage account>.table.core.windows.net/<table> * Azure Cosmos DB Table API accounts format: http://<storage account>.table.cosmosdb.azure.com/<table> |
| **TABLE** | A table is a collection of entities. Tables don't enforce a schema on entities, which means a single table can contain entities that have different sets of properties. |
| **ENTITY** | An entity is a set of properties, like a database row. **An entity in Azure Storage can be up to 1MB in size. An entity in Azure Cosmos DB can be up to 2MB in size** |
| **PROPERTIES** | * A property is a name-value pair. * Each entity can include up to 252 properties to store data. * Each entity also has three system properties that specify a partition key, a row key, and a timestamp. **Entities with the same partition key can be queried more quickly, and inserted/updated in atomic operations.** * An entity's row key is its unique identifier within a partition. |

#### ADDING DATA TO TABLE

|  |  |
| --- | --- |
|  | * For table storage – we can create tables in the azure storage account. * To add data to the table 🡪 Navigate to Storage Browser 🡪 Navigate to the table 🡪 Add Entity * An Entity has 2 parts   + Partition key   + RowKey |



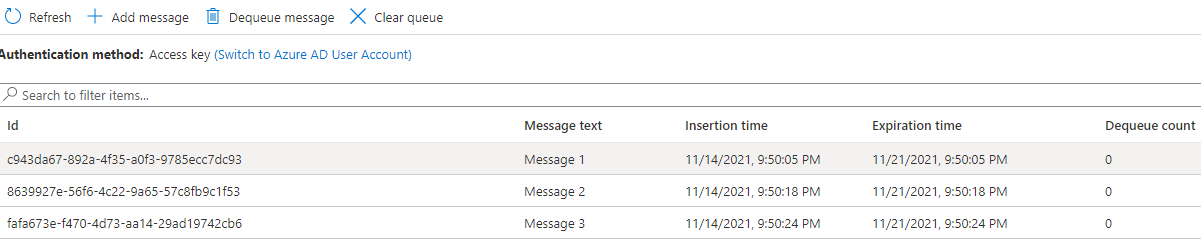
##### PARTITION KEY AND ROWKEY

* If we have huge amount of data in a table, dividing the table into partition – it becomes easier to search for an entity in particular partition.

### QUEUES

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

* Dequeue message will remove the message from top of the queue
* Clear queue will remove all the message of the queue.



Note in the queue

* The messages are added / dequeued from the queue programmatically.

#### EXAMPLE

