Contents

[AZURE - NETWORKING 3](#_Toc159343228)

[IP ADDESSS 3](#_Toc159343229)

[IPv4 ADDRESSING 3](#_Toc159343230)

[BINARY TO DECIMAL 3](#_Toc159343231)

[CLASSES IN IP ADDRESSING 3](#_Toc159343232)

[NETWORK ID IN IP ADDRESSING 4](#_Toc159343233)

[PUBLIC IP ADDRESS 4](#_Toc159343234)

[PRIVATE IP ADDRESS 5](#_Toc159343235)

[DIFFERENCE BETWEEN PUBLIC VERSUS PRIVATE IP 6](#_Toc159343236)

[SUBNETING 6](#_Toc159343237)

[SUBNET, SUBNET MASK AND CIDR NOTATION 8](#_Toc159343238)

[TYPES OF NETWORKING SERVICES 8](#_Toc159343239)

[VIRTUAL NETWORK 8](#_Toc159343240)

[COMPONENTS OF VNET 10](#_Toc159343241)

[CREATING A VIRTUAL NETWORK 10](#_Toc159343242)

[ADDING A NEW SUBNET 11](#_Toc159343243)

[PRIVATE SUBNET 12](#_Toc159343244)

[DEPLOYING VMs IN VIRTUAL NETWORKS 12](#_Toc159343245)

[EXTERNAL ACCESS 13](#_Toc159343246)

[OUTBOUND COMMUNICATION 13](#_Toc159343247)

[INBOUND COMMUNICATION 17](#_Toc159343248)

[IP ADDRESS ALLOCATION METHODS 17](#_Toc159343249)

[STATIC IP ADDRESS 17](#_Toc159343250)

[VIRTUAL NETWORK INTERFACE 19](#_Toc159343251)

[DEFAULT VIRTUAL NETWORK INTERFACE OF VM 20](#_Toc159343252)

[SECONDARY VIRTUAL NETWORK INTERFACE 20](#_Toc159343253)

[NETWORK SECURITY GROUP(NSG) 22](#_Toc159343254)

[NETWORK SECURITY GROUP OVERVIEW 22](#_Toc159343255)

[SETTING UP NSG FOR A VM ON NIC LEVEL 22](#_Toc159343256)

[SETTING UP NSG FOR A VM ON SUBNET LEVEL 25](#_Toc159343257)

[HOW NETWORK SECURITY GROUPS FILTER NETWORK TRAFFIC 26](#_Toc159343258)

[INBOUND TRAFFIC 27](#_Toc159343259)

[OUTBOUND TRAFFIC 27](#_Toc159343260)

[APPLICATION SECURITY GROUP 29](#_Toc159343261)

[SETTING UP APPLICATION SECURITY GROUP 29](#_Toc159343262)

[CONNECTING VIRTUAL NETWORKS 30](#_Toc159343263)

[COMMUNICATION BETWEEN VIRTUAL NETWORKS 30](#_Toc159343264)

[VIRTUAL NETWORK PEERING 31](#_Toc159343265)

[VNET GATEWAY (VNET TO VNET CONNECTION USING GATEWAY DEVICE) 34](#_Toc159343266)

[CONNECTING TO - ON PREMISE 38](#_Toc159343267)

[VPN GATEWAY 38](#_Toc159343268)

[EXPRESS ROUTE 40](#_Toc159343269)

[TYPES OF ENDPOINTS 41](#_Toc159343270)

[PUBLIC ENDPOINT 41](#_Toc159343271)

[SERVICE ENDPOINT 42](#_Toc159343272)

[PRIVATE ENDPOINT 43](#_Toc159343273)

[SERVICE ENDPOINTS AND SERVICE ENDPOINT POLICIES 47](#_Toc159343274)

[CONFIGURING SERVICE ENDPOINT 48](#_Toc159343275)

# AZURE - NETWORKING

## IP ADDESSS

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



|  |  |
| --- | --- |
|  | * The IP address has 2 parts * The Network part * 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 2 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*** | * Equivalent decimal: 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.

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

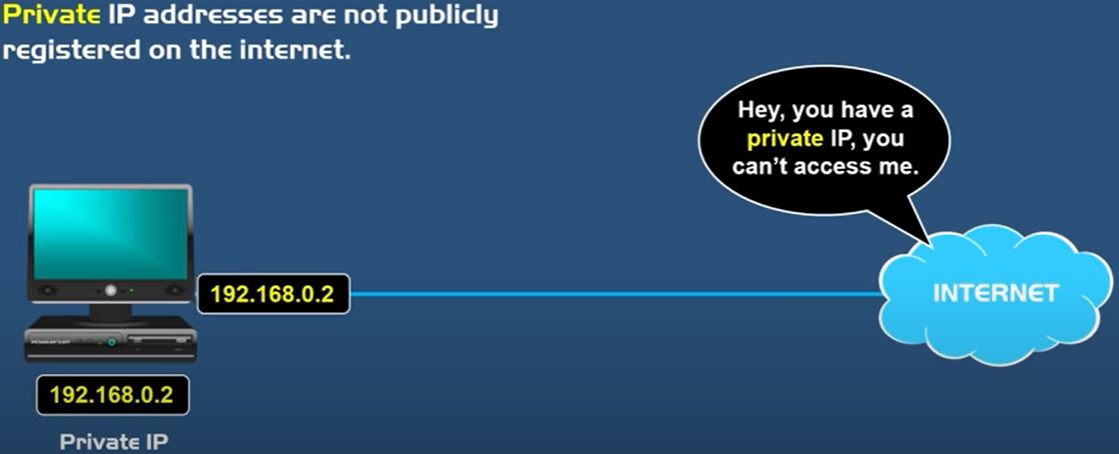
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CLASS** | **IP RANGE** |  | **NUMBER OF NETWORK** | **NUMBER OF HOST** |
| Class A |  | 1st 8 bits are Network Id  Remaining 24 bits are for Host | 126 | 2**24** = 16777216  16777216 – Network Id – Broadcast id =16777216 |
| Class B |  | 1st 16 bits are Network Id  Remaining 16 bits are for Host | 2**7** = 16,384 | 2**16** = 65536-2 = 65534 |
| Class C |  | 1st 24 bits are Network Id  Remaining 8 bits are for Host | 2**21** = 2,097,152 | 2**8** = 256-2 =254 |

|  |  |
| --- | --- |
|  | For the network part of the IP  For Class A - the 1st bit always “0”  For Class B - the 1st 2 bits always “10”  For Class C - the 1st 3 bits always “110”  CALCULATION OF NETWORK ID (EXAMPLE CLASS B)  Since the 1st 2 bits are always 1and 0 hence number of variable bits are 214 = 16384 |
|  | In the Host Part of IP. One Id is has been given to network ID and last Id will be broadcast ID |

### PUBLIC IP ADDRESS

|  |  |
| --- | --- |
|  | **WHAT IS A PUBLIC IP ADDRESS?**   * A public IP address is a unique identifier assigned to a device (such as a computer or router) that is connected to a network, specifically the internet. * It allows devices to communicate with other devices and services on the internet. * Public IP addresses are provided by Internet Service Providers (ISPs) and can be either dynamic (changing periodically) or static (fixed and unchanging). |

### PRIVATE IP ADDRESS



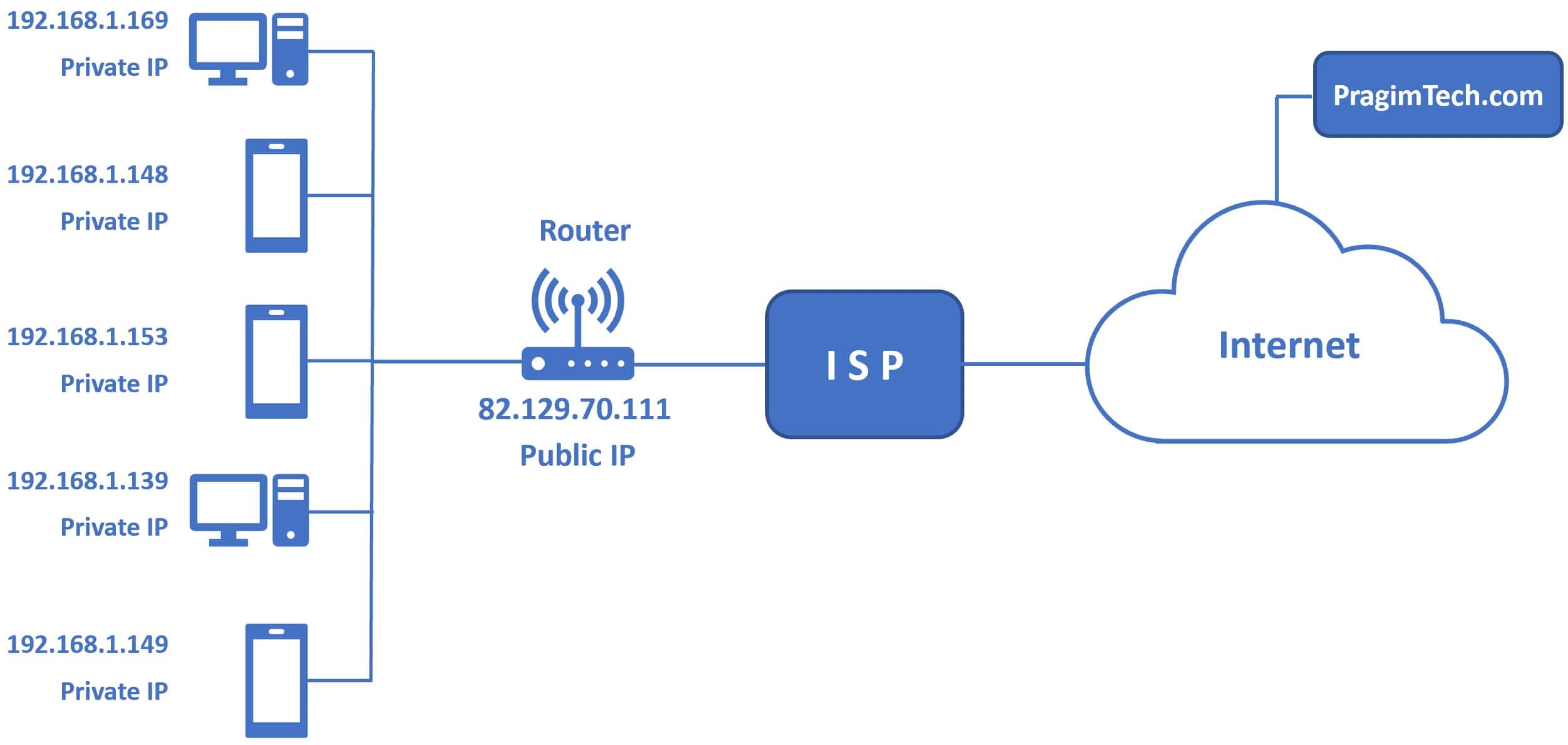
**WHAT IS A PRIVATE IP ADDRESS?**

* A private IP address is the address which a network router assigns to the connected devices. **Each device within the same network is assigned a unique private IP address**.
* The devices communicate to each other through private IP address, without connecting to internet.
* **DHCP (***Dynamic Host Configuration Protocol***) is a service used in the routers to assign private to the devices.**
* The devices withing Home/office network- can access the internet using NAT***.*** When a device on a private network sends a request to access the internet, the NAT router modifies the source IP address of the outgoing packets to its own public IP address. This allows the device to communicate with the internet without revealing its private IP address. When the response is received, the NAT router translates the destination IP address back to the private IP address of the requesting device and forwards the response to it. ***These services are built-in the Router.***

**PRIVATE IP RANGE**



#### COMMUNICATION WITH INTERNET



HOW TO FIND PRIVATE IP ADDRESS ?

|  |  |
| --- | --- |
| 1. Run window 🡪 type **ipconfig** 2. **In the output, we should see is the IPv4. This is the private IP address of the device.** |  |

HOW TO FIND PUBLIC IP ADDRESS?

1. Type **ipconfig** in command prompt 🡪 Find the default gateway IP address.
2. Type the default gateway IP address in the browser (http://192.168.1.254)
3. This takes you to the admin router page and you should see the WAN IP address here.
4. WAN IP address is your public IP address. WAN IP is just another term for public IP

### DIFFERENCE BETWEEN PUBLIC VERSUS PRIVATE IP

|  |  |
| --- | --- |
| PUBLIC IP ADDRESS | PRIVATE IP ADDRESS |
| External (global) reach | Internal (local) reach |
| Used for communicating outside your private network, over the internet | Used for communicating within the private network, with other devices in your home or office |
| A unique numeric code never reused by other devices | A non-unique numeric code that may be reused by other devices in other private networks |
| Found by Googling: "What is my IP address?" | Found via your device’s internal settings |
| Assigned and controlled by your internet service provider | Assigned to your specific device within a private network |
| Not free | Free |
| Any number not included in the reserved private IP address range  Example: 8.8.8.8. | IP RANGE   * CLASS A - 10.0.0.0 — 10.255.255.255 * CLASS B - 172.16.0.0 — 172.31.255.255 * CLASS C- 192.168.0.0 — 192.168.255.255   Example: 10.11.12.13 |

### SUBNETING

* **Subnet is a network withing a network.**
* Subnetting is a concept of using the IP address efficienty

#### SUBNET MASK

|  |  |
| --- | --- |
|  | **CALCULATION OF SUBNET MASK (CLASS A)** |

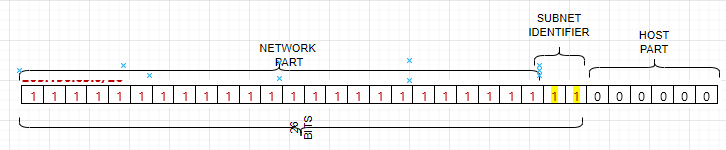
#### CIDR NOTATION (CLASSLESS INTER DOMAIN ROUTING)

* **CIDR** gives the number of bits given to the network bit. For example, **192.168.100.1/24** This represent that out of 32 bits the first 24 bits are used for network id. It is also called network pre-fix.

EXAMPLE

**205.150.65.0/26**

1. The IP belongs to Class C
2. By default, in Class C - 24 bits are assigned to network and remaining 8 bits host. Hence the CIDR notation for Class C address in “24“
3. But in above example – since 2 more bits has been taken from Host. Hence the CIDR notation will be “26”



WHY WE DO THIS? – This help in subnetting.

#### SUBNETING CALCULATONS

Question: If Network Address Is – 205.150.65.0/26.

**SUBNET MASK**

* + Since it a class C IP. The default subnet mask is 255.255.255.0. But since we borrowed 2 more bits from the host bit hence the new subnet mask will be – 255.255.255.192 (*2^7 + 2 ^6 = 192*)

**NETWORK ID**

* + It’s a logical AND operation of Network address and Subnet mask – **205.150.65.0**
  + **NUMBER OF POSSIBLE SUBNETS =** 2**(number of subnet identifier)** = 22 = 4 Subnets
  + **NUMBER OF POSSIBLE USABLE HOST (IN EACH SUBNET) =** 2**(number of host bits)** -2 = 26 -2 = 64-2 = 62 Hosts
    1. 2 is subtracted because 1st IP is reserved for Network and last IP for broadcast.

1. **BROADCAST IPs (IN EACH SUBNET)**
   * This is the last IP in SUBNET. So, it will be
   * Broadcast Id for Subnet 0 - 205.150.65.63
   * Broadcast Id for Subnet 1 - 205.150.65.128
   * Broadcast Id for Subnet 2 - 205.150.65.192
   * Broadcast Id for Subnet 3 - 205.150.65.256

Question: If Network Address Is – 205.150.65.0/24. Create 10 Subnet from this network address

* This is a Class C IP address. If we have to create 10 subnets – Then we have to first decide how many bit, we will be borrowing from host bits. To create minimum 10 subnets, it will be 24 = 16 (4 bits we need to borrow to accommodate 10 subnets)

|  |  |
| --- | --- |
| SUBNET MASK | 255.255.255.240 |
| NETWORK ID | 205.150.65.0 |
| NO OF SUBNET | 24 = 16 Subnets |
| NO OF HOST (PER SUBNET) | 24-2 = 14 Hosts per subnet |
| BROADCAST ID | 205.150.65.15 |

### 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?

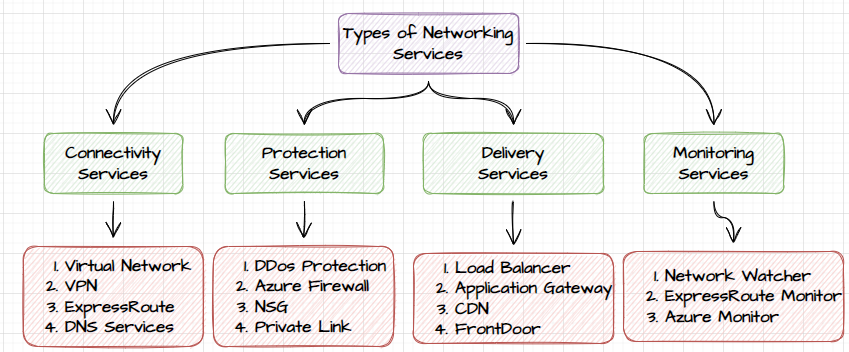
* Since the IP is consist of 2 parts - Network and Host part. Network Part in a network will be same for all host.
* Host decide the network Id with the help of Subnet Mask.
* Subnet mask is also of 32 bits -which has a mapping with the IP address. The 1s represent the network portion and 0s are the 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

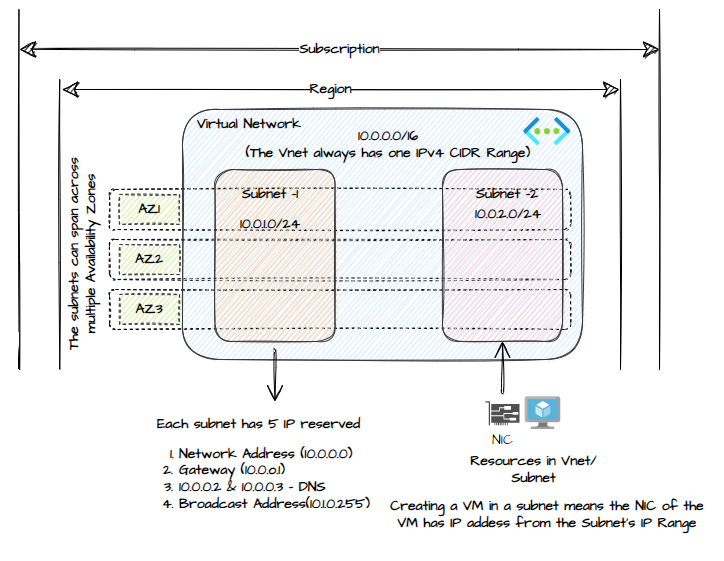
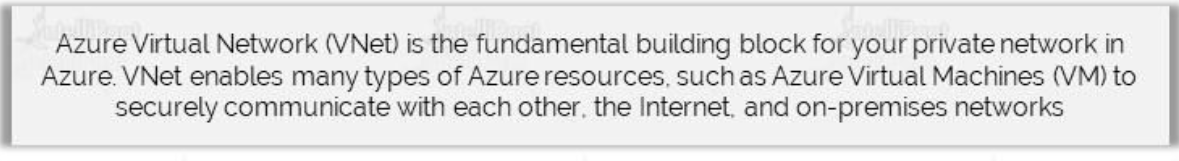
|  |  |
| --- | --- |
| **150.10.20.30** | * The IP belongs to Class B * Network Id: 150.10.0.0 * Broadcast ID: To calculate the broadcast id, set the host part of Network to 255. Hence the broadcast Id - 150.10.255.255 * Usable Host IP = Total Number of IP address – (Network IP + Broadcast IP) i.e.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

## TYPES OF NETWORKING SERVICES



# VIRTUAL NETWORK

* Azure virtual network allow us to create an isolated /private network on cloud. Example- Any network created for Azure account will be solely dedicated for that account itself.



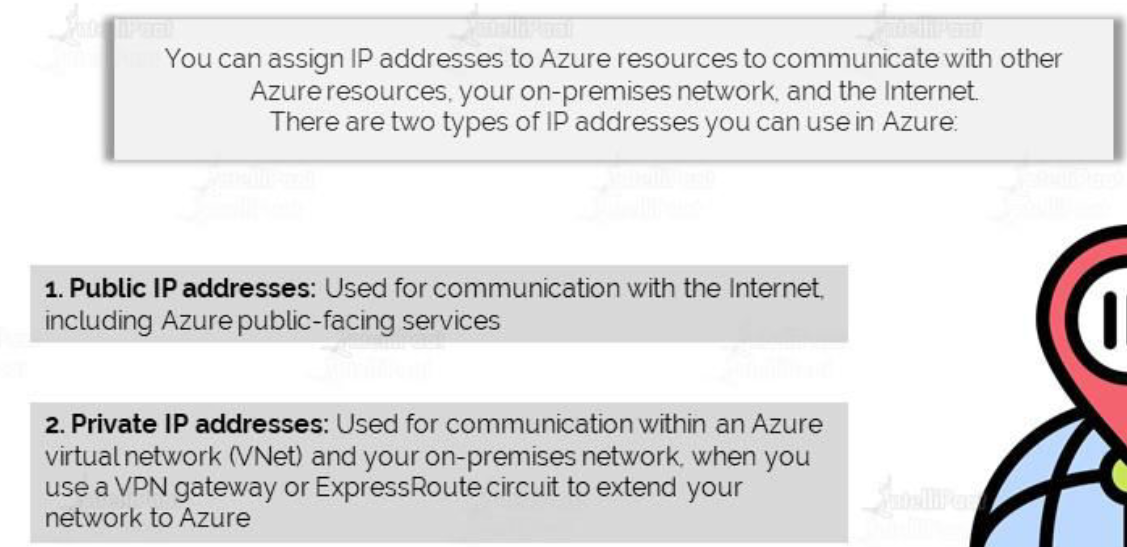
* The Gateway IP address enables the communication between the resources in different subnets.

* The Virtual Network exists.
  + *Within a specific subscription*
  + *Within a specific region*
  + *It cannot span subscriptions nor regions.*
* When we create a virtual network
  + Specify an IP address range.
  + With the help of the IP address range, we can define subnets, where they also have an IP address range that is a subset of the IP address range of the virtual network.
  + When the VMs, which gets deployed as part of the subnet, gets a private IP address assigned to the network interface and the private IP address that gets assigned depends upon the IP address range of the subnet itself.

|  |
| --- |
| * IF WE WANT TO ISOLATE THE WORKLOADS THAT RUN ON VIRTUAL MACHINES, THEN DEPLOY THEM IN A DIFFERENT VIRTUAL NETWORKS. * BY DEFAULT, OUTBOUND TRAFFIC TO THE INTERNET FROM THE VIRTUAL NETWORK IS ALLOWED. |

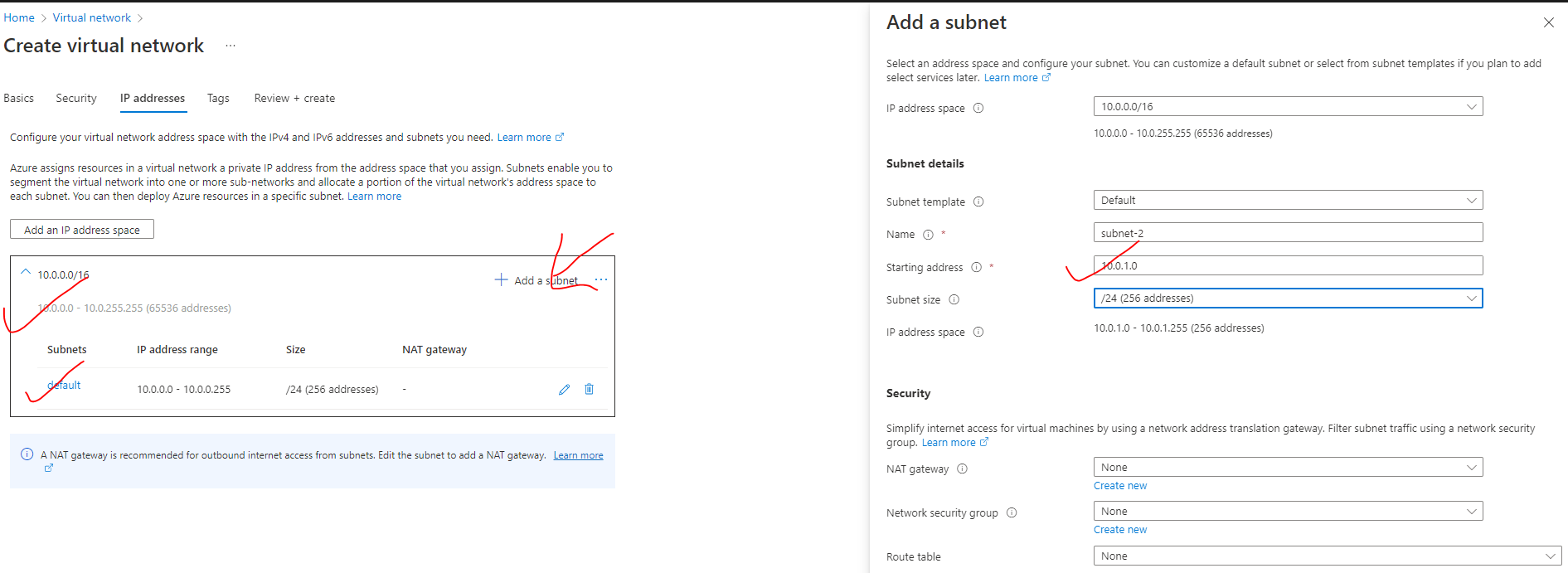
## COMPONENTS OF VNET

|  |  |
| --- | --- |
| **ADDRESS SPACE** | * Address Space is the range of private IP addresses for Virtual Network * Azure will assign the next available IP address from the address space to the resources in the Virtual Network |
| **SUBNETS** | * A subnet is a logical segment within the Virtual network. * A subnet is allocated a portion of a Virtual Network’s address space. |
| **REGION** | * Virtual Network are scoped to a single region. * Multiple virtual networks from different region can be connected using VNET peering |
| **SUBSCRIPTION** | * Virtual Network are scoped to a subscription. * We can implement multiple VNETs within each subscription and Azure region |



## CREATING A VIRTUAL NETWORK

* **VNET creates a LAN – where all the associated resources get a private IP**. Creating such private network help in managing the in and out traffic.
* When a VNET is created in Azure – by default it will have internet access - Hence all the associated VMs will have internet access too, whether the VM has public IP or not.

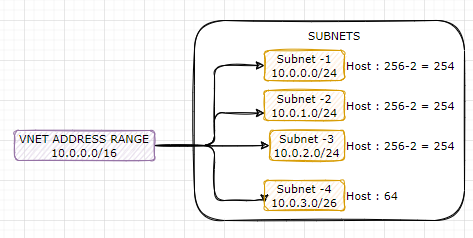


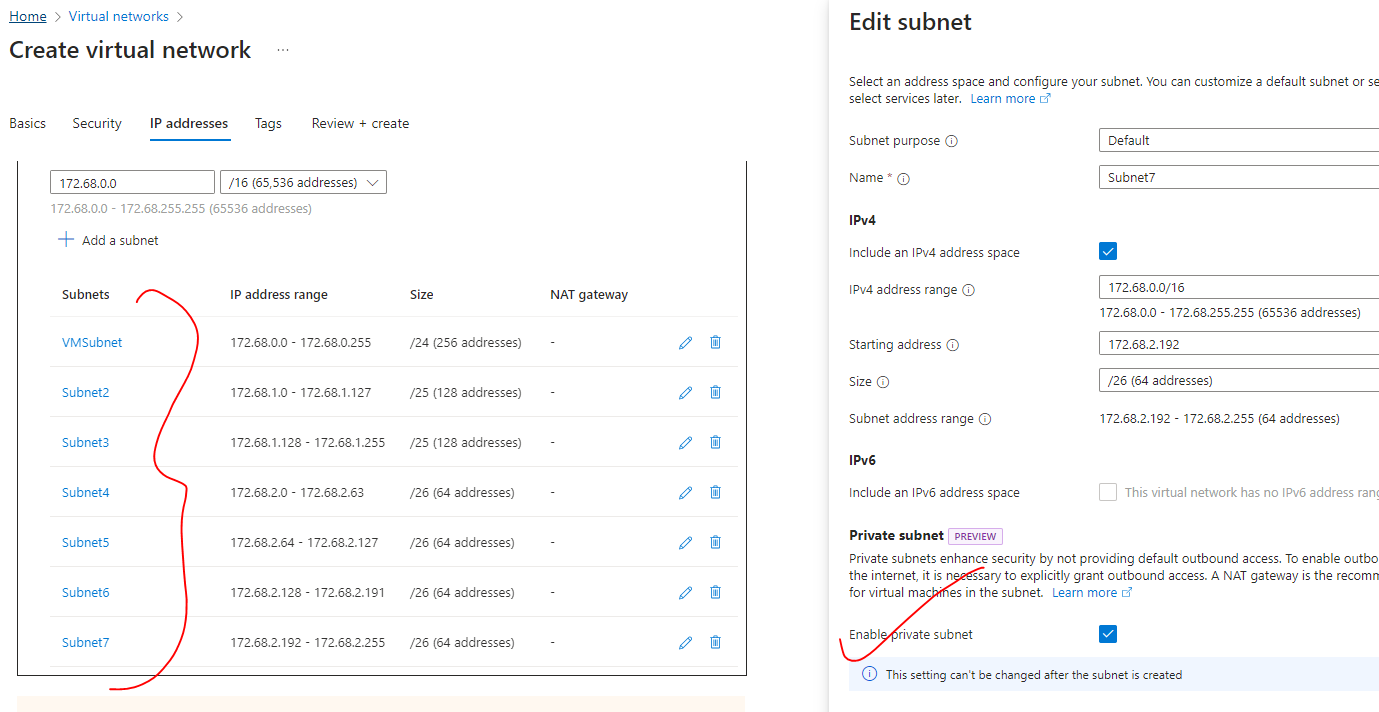
**IP ADDRESS**

* As per the above configuration (default configuration)- The VNET has 65536 addresses. Since VNET always set up a LAN in a region – hence all the IP will be Private IPs.
* By default, it creates a subnet of 10.0.0.0/24 – of 256 addresses.

### ADDING A NEW SUBNET

* <https://www.davidc.net/sites/default/subnets/subnets.html?network=10.0.0.0&mask=20&division=23.f42331>
* When the subnet is created 5 addresses are reserved for in each subnet
* **1 FOR DEFAULT GATEWAY**
* **2 FOR DNS**
* **1 FOR NETWORK ID (10.0.0.0)**
* **1 FOR BROADCAST (10.0.0.255)**
* ***Hence if the IP range of a subnet start from 10.0.0.0 , then the 1st VM will get the IP of 10.0.0.4***







**For each subnet – Available IP address = Total IPs in Subnet – (1 Network Id + 2 DNS + 1 Broadcast +1 Default Gateway)**

***Note : The subnet can be deleted only if we don’t have any resource associated within that subnet.***

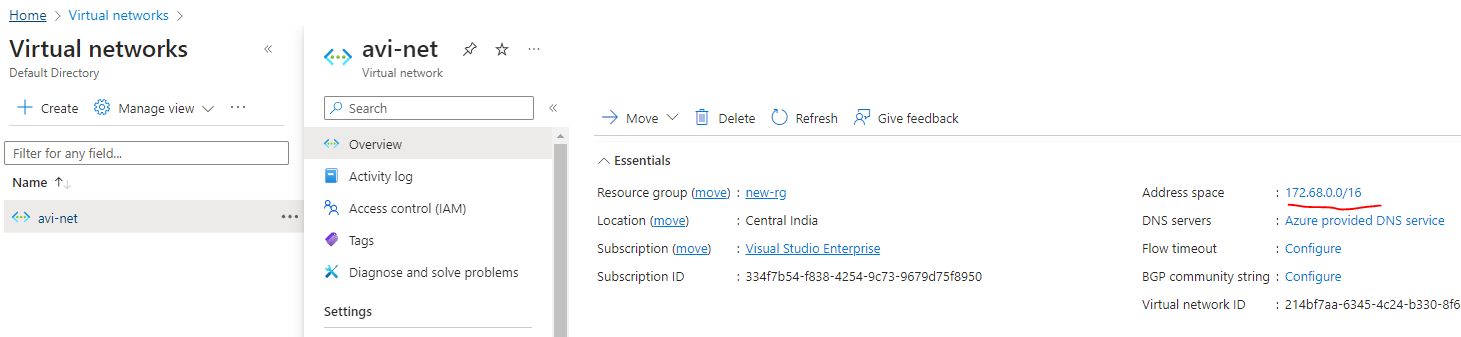
|  |  |
| --- | --- |
|  | * The VM associated with different subnets within the VNET can be able to communicate with each other via their private IP. There is no need of public IP address for the communication with the VNET. * **This happens because Azure enables the routing, by default between the subnets** |

### PRIVATE SUBNET

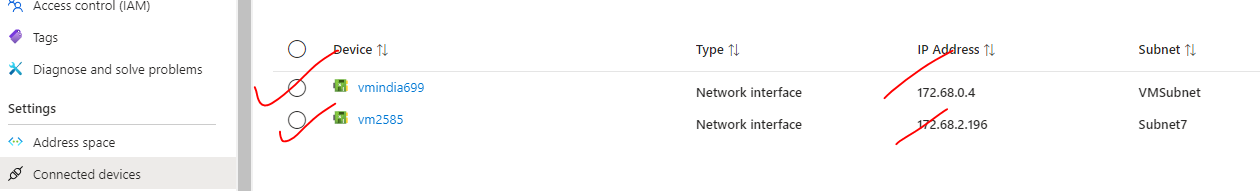
* While creating a subnet in a VNET we can mark/ enable the subnet as “Private Subnet”. A private subnet is a range of IP addresses that are reserved for use within a private network.
* These IP addresses are not routable over the internet and are typically used for internal communication within a specific organization or network.
* Private subnets provide a level of security by isolating internal network traffic from external sources.
* **They are commonly used in conjunction with Network Address Translation (NAT) to allow devices within the private subnet to access the internet using a single public IP address. (More details below..)**

## DEPLOYING VMs IN VIRTUAL NETWORKS

* ***To associate a VM to a given virtual network. The virtual machine must be created in in the same region as of VNET.***
* In the below example –the IP address, assigned to VM start with **172.68.0.4** because *first 5 Ips are reserved for default gateway -1, network id -1, broadcast id -1, DNS- 2*



* The VM (the list shows the NIC associated with the VNET) will show up as a Connected device in the Virtual network.
* Note: Until a VM is connected to a subnet – the subnet cannot be deleted from the VNET.



## EXTERNAL ACCESS

### OUTBOUND COMMUNICATION

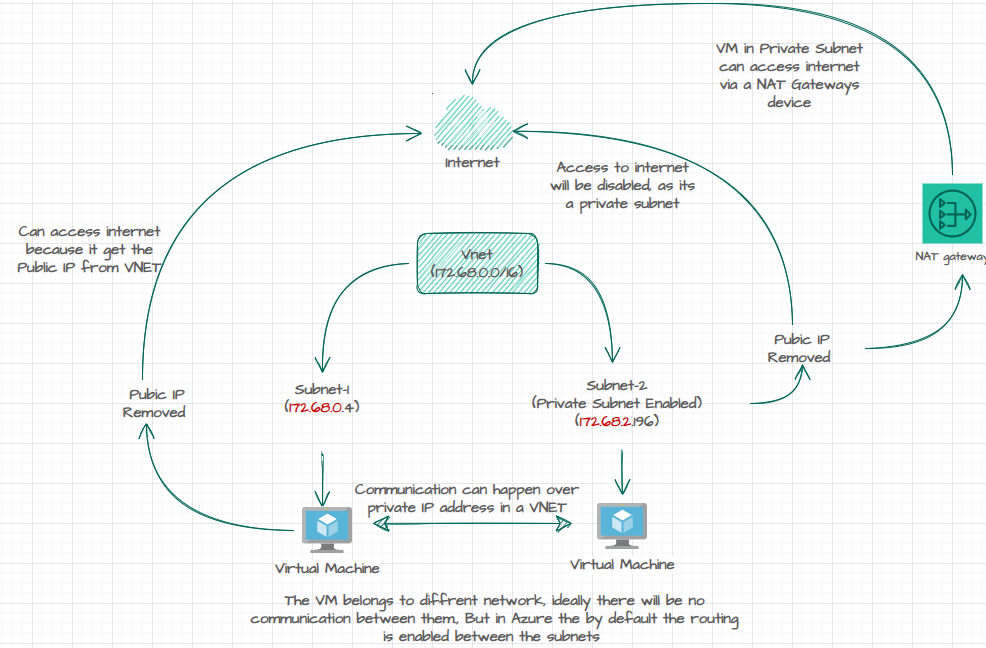
#### FOR RESOURCES IN SUBNET

* The virtual machines within the Virtual network can communicate using their private IP address.
* The resources in a VNET can by default access the internet using SNAT(Source Network Address Translation).
* ***SNAT (Source Network Address Translation) is a mechanism used to translate the source IP addresses of outgoing packets from resources within the VNet to a single public IP address. It enables resources within the VNet to communicate with external networks, such as the internet, while using a consistent public IP address.***

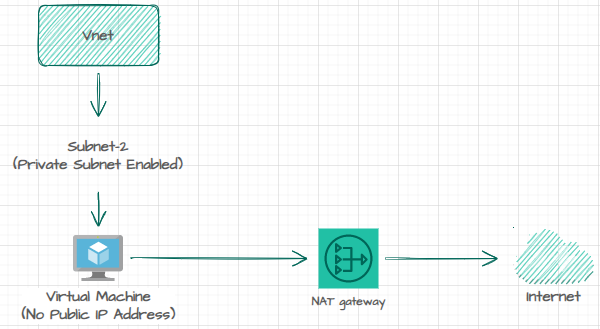
|  |  |
| --- | --- |
|  |  |

#### FOR RESOURCES PRIVATE SUBNET

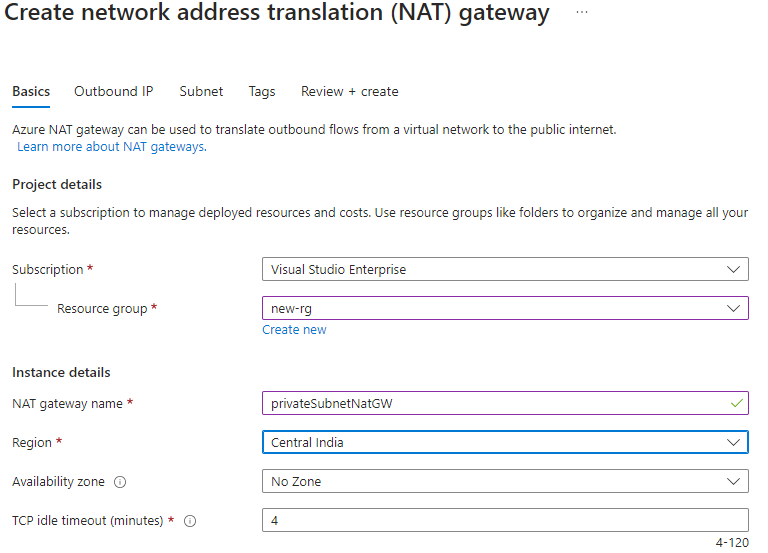
* If the VM in private subnet does not have a pubic IP address cannot able to access internet. It can able to access internet only via NAT gateway device.



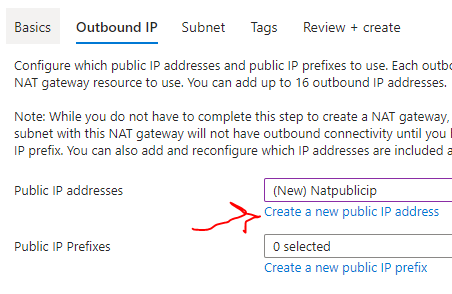
1. Create a VNET 🡪 add a Subnet and enable the private subnet.
2. Deploy the VM in the same subnet without any public IP address.



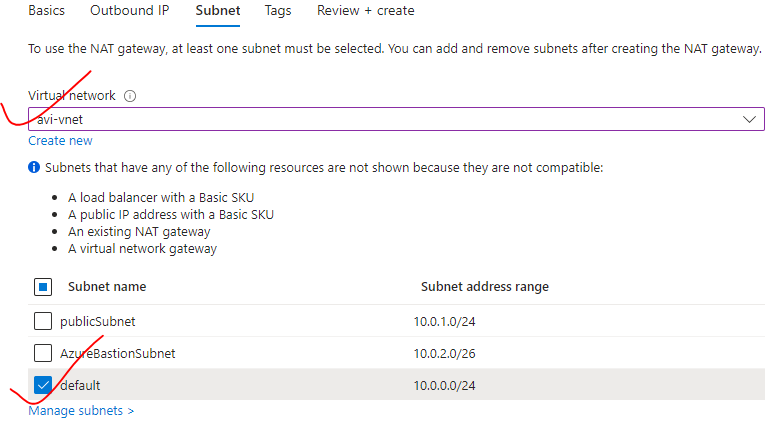
##### DEPLOYING NAT GATEWAY



CREATE A PUBLIC IP FOR EXTERNAL COMMUNICATION

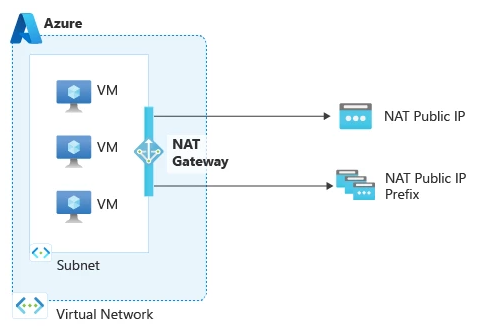


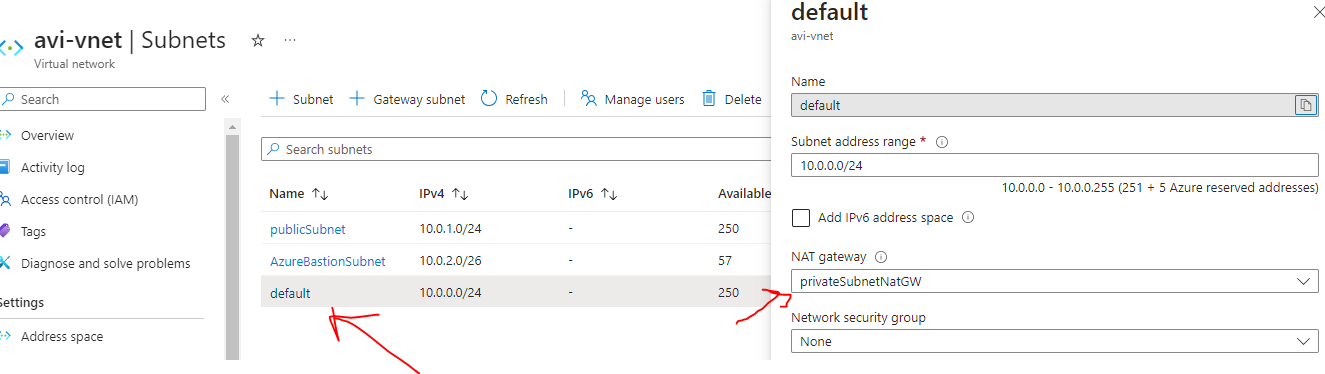
SELECT THE PRIVATE SUBNET



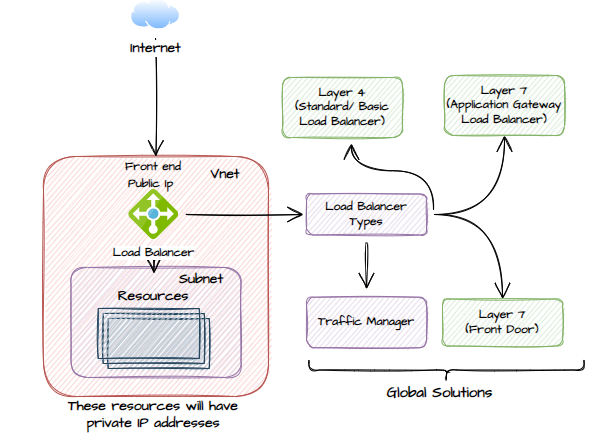
THE SUBNET WILL HAVE THE NAT GATWAYS DEVICE ATTACHED

* Once the NAT Gateway device is attached to subnet – it can be able to access internet via the public IP address of the NAT gateways. i.e. the NAT gateway allows the outbound connection to external network.





### INBOUND COMMUNICATION



## IP ADDRESS ALLOCATION METHODS

* We can assign IP address to Azure resource to communicate with other Azure resources.

|  |  |
| --- | --- |
| **PUBLIC IP** | * Used for the communication within the internet, including Azure public facing services |
| **PRIVATE IP** | * Used for the communication within the Azure VNET and On-Premises Network, when we use a VPN gateway or ExpressRoute circuit to extend network in Azure * A private IP address is assigned to the resource in VNET. The addresses are assigned from the address pool range of VNET. * The resources can be able to communicate with each other using their private IP address. |

**THERE ARE 2 METHODS OF IP ADDRESS ALLOCATION**.

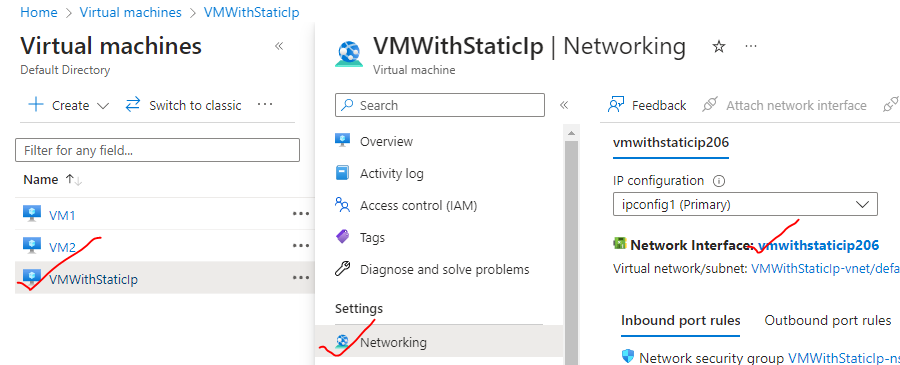
|  |  |
| --- | --- |
| DYNAMIC ALLOCATION | * Dynamic IP allocation involves automatically assigning IP addresses to devices or resources using DHCP (Dynamic Host Configuration Protocol). * DHCP servers in the network dynamically assign available IP addresses from a predefined pool to devices as they connect to the network. * Dynamic IP addresses are temporary and can change over time. When a device reconnects to the network, it may receive a different IP address from the DHCP server. * Dynamic IP allocation is commonly used for devices like laptops, desktops, smartphones, and other devices that connect to the network temporarily or frequently. * DHCP also provides additional configuration information, such as subnet mask, default gateway, and DNS server addresses, along with the IP address. |
| STATIC ALLOCATION | * Static IP allocation involves manually assigning a specific IP address to a device or resource. * A static IP address remains constant and does not change unless manually reconfigured. * Static IP addresses are typically used for devices or resources that require a fixed, predictable address, such as servers, network printers, or network appliances. * When using static IP allocation, administrators need to ensure that IP addresses are unique and properly managed to avoid conflicts within the network. |

### STATIC IP ADDRESS

|  |  |
| --- | --- |
|  | * When we create and associate a VM, it gets a public IP address and private IP address. All of these are dynamic IPs. * When we stop and the start the VM, it will get allocated with a new public IP address * This can be an issue when, let's say that we have a Web application hosted on the VM. To make it accessible to the user using some DNS (e.g., **cloudportalhub.com**), the external DNS provider map the domain name with the public IP address of the VM. * If the VM is stopped and started again, the IP address the changes and if the IP address changes, this link is broken so users will not be able to reach the application. * To mitigate this issue, we can allocate a static IP address to both the public IP and private IP address. |

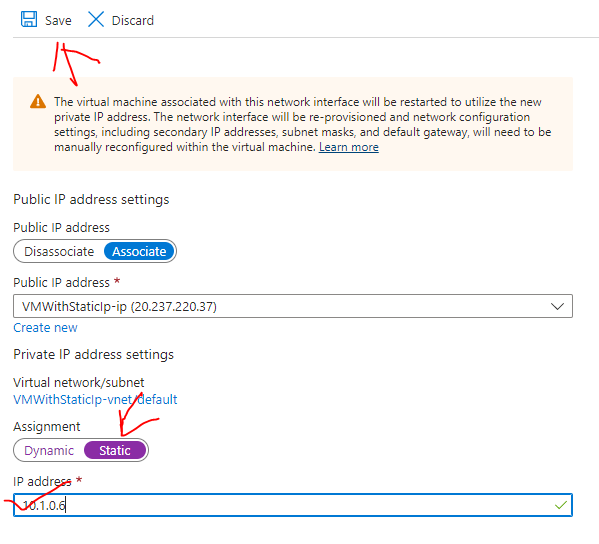
#### ASSIGNING STATIC IP ADDRESS

To assign static IP address to a VM

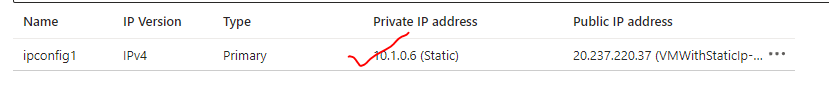




* Since it an static IP address – We can assign an available IP addess manually (from the address pool of the VNET)



STATICALLY ASSIGNED IP ADDRESS TO A VM



## VIRTUAL NETWORK INTERFACE



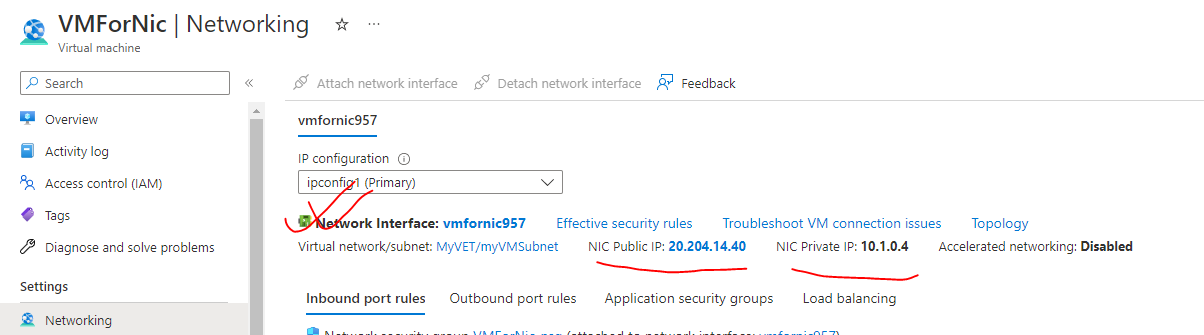
|  |  |
| --- | --- |
|  | * The network interface is a physical device is used as an interface between the machine (VM) and Network (VNET). In Azure. * As the VMs are in the cloud, they are known as a ***virtual network interface***. * *Any device (Physical or Virtual) which need to connect to internet needs a Network Interface Card (It can be wired or wireless).* ***Every VM in Azure has virtual network interface that is attached to it by default****. It gets assigned to VM while creating a VM itself.* * **The NIC are then attached to VNET/Subnet** * A VM must have one NIC - but more NIC can also be attached VM-depending upon on the size of the VM itself. |

* THE VIRTUAL NETWORK INTERFACE CARD GETS BOTH **PRIVATE AND PUBLIC IP ADDRESS**

|  |  |
| --- | --- |
| PRIVATE IP ADDRESS | PUBLIC IP ADDRESS |
| * The private IP of VNI is used for the communication within the VNET. * The private IP lies in the range of Subnet of which the VM is part of | * The public IP is used for the communication on internet. * This is assigned by Azure itself |

### DEFAULT VIRTUAL NETWORK INTERFACE OF VM

* A default VNIC get attached to a VM during its creation. The VNIC has both private and public IP



VNIC PROPERTIES



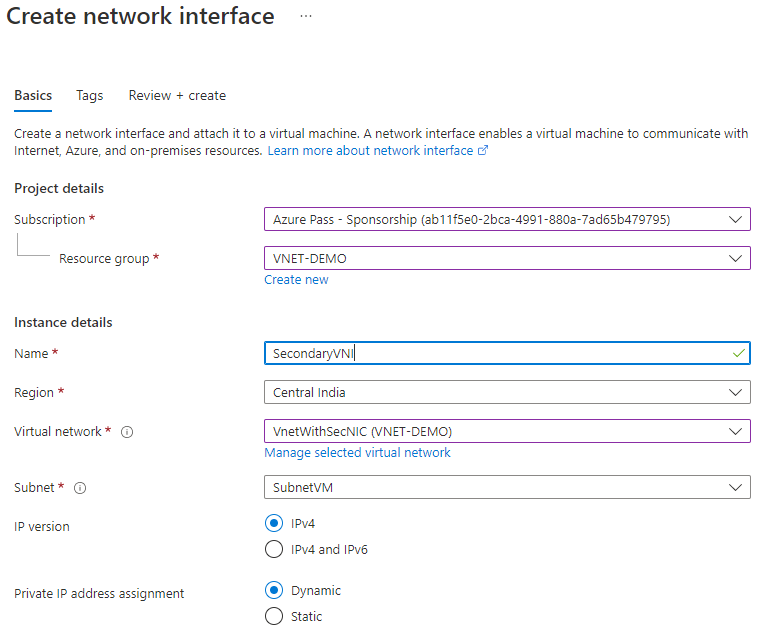
### SECONDARY VIRTUAL NETWORK INTERFACE

* By default, the VM has a Virtual Network Interface attached to it – But we can attach a secondary VNI to a VM

#### CREATING A SECONDARY VIRTUAL NETWORK INTERFACE

ADDING A SECONDARY V-NIC

|  |  |
| --- | --- |
|  | **WHY WE NEED A SECONDARY VNI?**   * + When a VM is created – it is attached to a default subnet within a Virtual Network.   + The V-NIC help in the communication of VM with the subnet.   + Adding a secondary NIC to a VM helps the VM to connect with another subnet within the Virtual network.   The |
|  | **USE-CASE**   * + Let’s say we have traffic coming from Backend and front end.   + We can have secondary subnet for each traffic to avoid network congestion. |



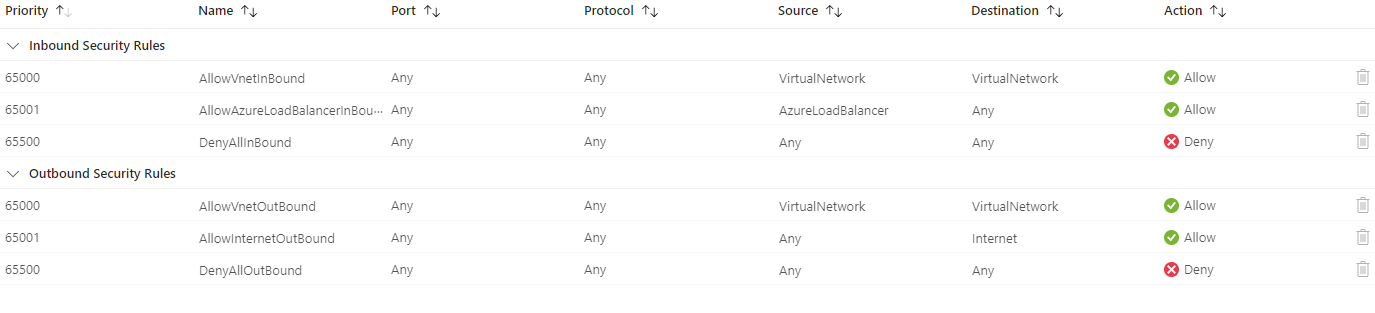
ATTACH VNIC TO VM

|  |  |
| --- | --- |
|  | * + To attach a Network interface to a VM – we need to first STOP the VM(Deallocated)   + We can connect network interfaces in the same VM to different subnets within a virtual network. However, the network interfaces must all be connected to the same virtual network. |
|  |

## NETWORK SECURITY GROUP(NSG)

|  |  |
| --- | --- |
|  | 1. NSG is a separate resource in Azure . The NSG is attached it to the network interface. The network interface in turn is attached onto a virtual machine. 2. A NSG is used to filter incoming and outgoing traffic to and from several types of Azure Resources. 3. The purpose of network security groups is to manage (Deny/ Allow) the traffic that's coming in and going out of Azure Resources 4. It has rules for inbound (coming into the VM) and outbound (going out of the VM) traffic. 5. We can have multiple NSG rules, which has the priority associated with it. The rules are evaluated in the order of the priority. |

**DEFAULT NSG RULE**



INBOUND RULES

|  |  |
| --- | --- |
| **AllowVnetInBound** | * + The "AllowVnetInBound" NSG (Network Security Group) rule allows incoming network traffic from within the Virtual Network (VNet). (Check the source and destination) |

OUTBOUND RULES

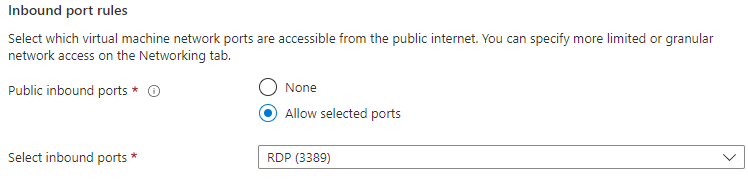
|  |  |
| --- | --- |
| **AllowVnetOTBound**  **AllowInternetOutBound** | * + All outbound traffic from VNET is allowed |

### NETWORK SECURITY GROUP OVERVIEW

|  |  |
| --- | --- |
| NSG-processing | * NOTE – * The NSG cannot be attached to VNET . The NSG can be applied on 2 levels.  1. AT THE SUBNET LEVEL (OF WHICH THE VM IS PART OF) – This takes high priority 2. AT THE NETWORK INTERFACE CARD LEVEL OF THE VM |

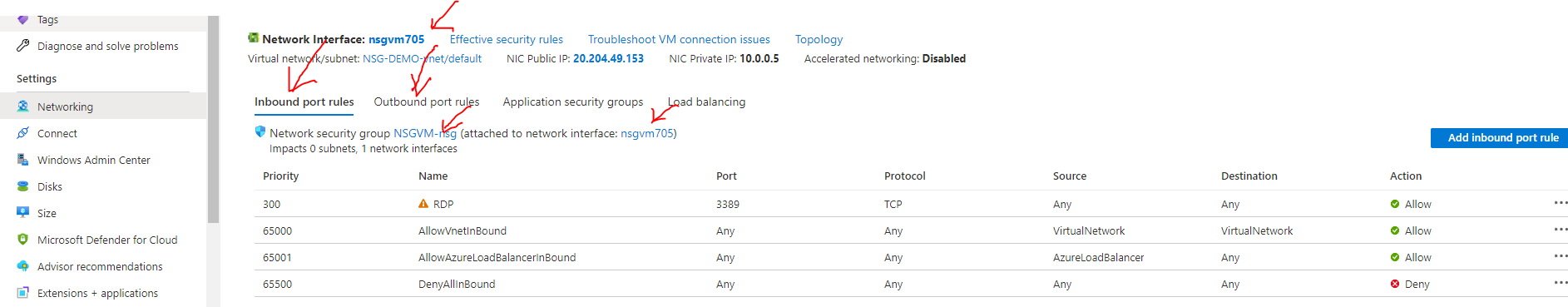
### SETTING UP NSG FOR A VM ON NIC LEVEL

* **STEP 1: CREATE A VM** – While creating a VM (in Basic and Networking tab we have setting available for NSG)– we set some “inbound port rules”- the actually creates an inbound NSG rule for the given VM



The NSG is a separate resource in Azure VM. In the below image – we can see that.

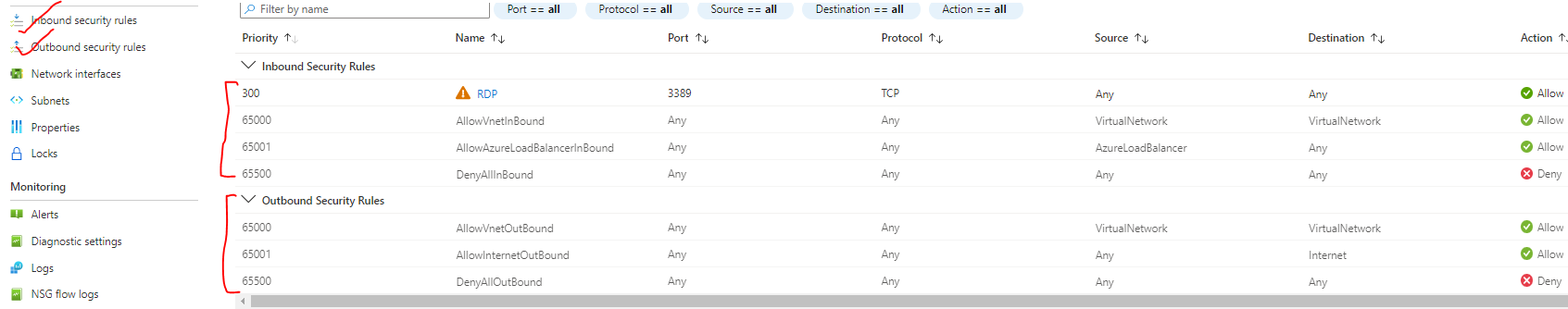
* The NSG can be viewed in Networking section of the VM
* The NSG has both Inbound and outbound rules for the traffic
* The NSG are attached to the NIC (Network interface). Note – we can also attach the NSG to the subnet (of which VM is part of) as well



#### NSG RULES

DEFAULT NSG RULES

* By default, NSG has some default inbound and outbound NSG rules. We cannot change the default NSG rules.

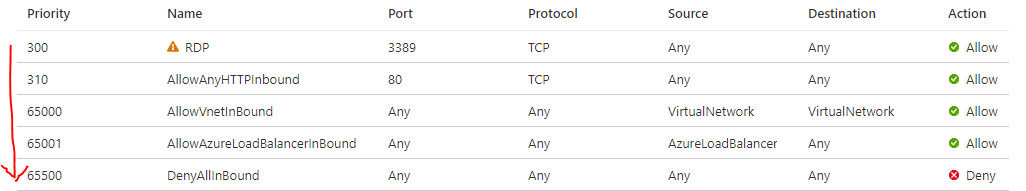


|  |  |
| --- | --- |
|  | **USE CASE FOR NSG**   * Let’s say we have IIS installed on VM – which we want to allow the access from any machine. * For this we need to add a NSG rule (inbound – as it’s an inbound traffic) |

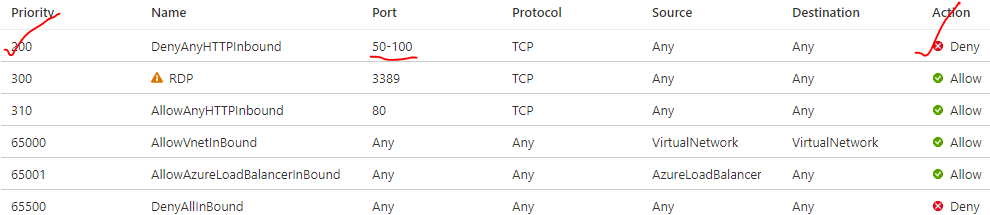
#### PRIORIRY SETTINGS IN NSG RULES

| PROPERTY | EXPLANATION |
| --- | --- |
| **NAME** | * A unique name within the network security group. |
| **PRIORITY** | * A number between **100 and 4096.** Rules are processed in priority order, with lower numbers processed before higher numbers, hence **lower numbers have higher priority**. * Once traffic matches a rule, processing stops. As a result, any rules that exist with lower priorities (higher numbers) that have the same attributes as rules with higher priorities aren't processed. |
| **SOURCE OR DESTINATION** |  |
| **PROTOCOL** | * TCP, UDP, ICMP, ESP, AH, or Any. The ESP and AH protocols aren't currently available via the Azure portal but can be used via ARM templates. |
| **DIRECTION** | * Whether the rule applies to inbound, or outbound traffic. |
| **PORT RANGE** | * We can specify an individual or range of ports. |
| **ACTION** | * Allow or deny |

* THE RULES ARE EXECUTED FROM TOP TO BOTTOM (low to high priority values)
* **IF THE CONDITION IS MEET FOR A CONFIGURED RULE – NO FURTHER RULES ARE EVALUATED**

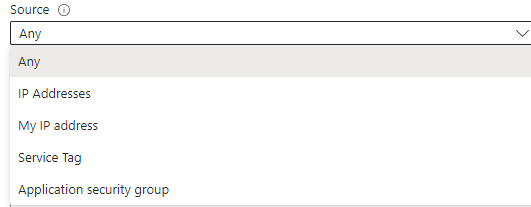


* *Example - In the below rules the Inbound to PORT 80 is denied. As the port “80” comes in the range of 50-100. Now once the rules in matched – no further evaluation of rules will be processed*



#### NSG– IP ADDRESS

SOURCE AND DESTINATION SETTINGS



When configuring NSG rules, we encounter the terms "source" and "destination," which are used to define the network traffic flow and specify the endpoints involved.

* **Source**:
  + The source refers to the origin of the network traffic.
  + It represents the IP address, IP range, or application security group from which the traffic originates.
  + The source can be defined based on various criteria, such as a specific IP address, an IP range (CIDR notation), a virtual network, a subnet, or even an application security group.
* **Destination**:
  + The destination represents the **target endpoint or resource to which the network traffic is directed**.
  + It can be defined using similar criteria as the source, such as a specific IP address, an IP range, a virtual network, a subnet, or an application security group.
* When creating NSG rules, we define both the source and destination to control the flow of traffic between them. For example, we can configure a rule that allows inbound traffic from a specific IP address range (source) to a particular virtual machine or subnet (destination). Conversely, you can create rules that restrict or deny traffic from certain sources to specific destinations.
* By specifying the source and destination in NSG rules, you can effectively manage and secure the network traffic flow within your Azure environment. It allows you to define granular access control policies and protect your resources from unauthorized access or malicious activities.

#### NSG - OUTBOUND RULES

#### NSG – ALLOW ICMP

* **ICMP stands for Internet Control Message Protocol**. It is a network protocol used for sending error messages and operational information about network conditions. ICMP is primarily used by network devices, such as routers, to communicate with each other and to report errors.
* It is also commonly used by network administrators for troubleshooting network connectivity issues. ICMP messages are typically carried within IP packets and can be used for various purposes, such as checking if a host is reachable, determining network congestion, or diagnosing network problems. E.g ping command

### SETTING UP NSG FOR A VM ON SUBNET LEVEL

* As the NSG rules can be applied to NIC and Subnet level (Subnet of which the VM is part of). In this case the NSG rule at subnet level will be evaluated first for NIC rules.

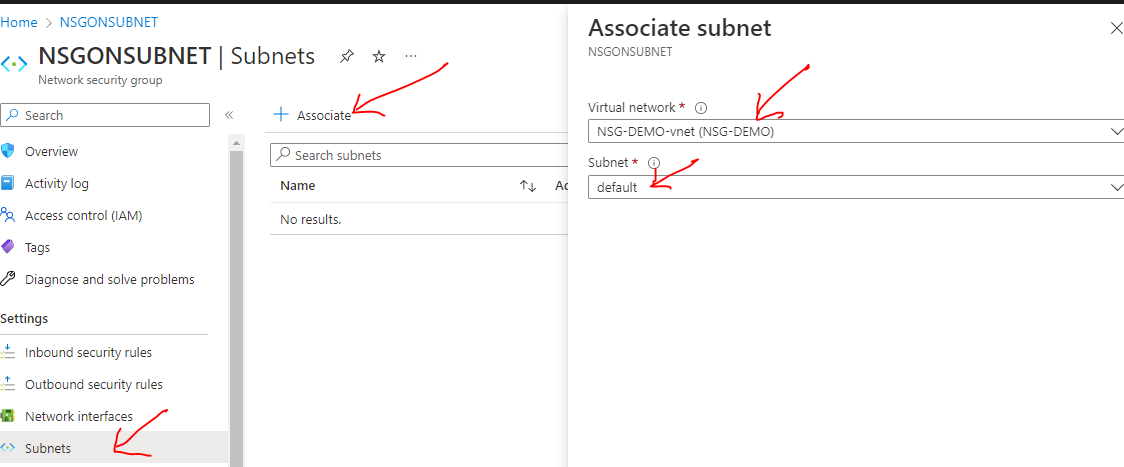
**TO CONFIGURE NSG ON SUBNET LEVEL**

Step 1: CREATE AN NSG –

* NSG is a separate resource in Azure. The NSG must be created in the same region of Virtual Network of the Virtual Machine.

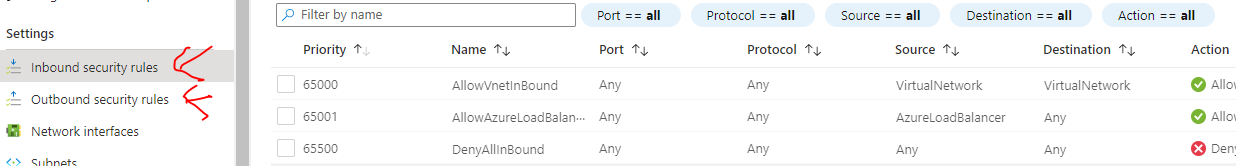
|  |  |
| --- | --- |
| VNET OF THE VM |  |

Step 2: NAVIGATE TO THE NSG AND ASSOCIATE THE SUBNET

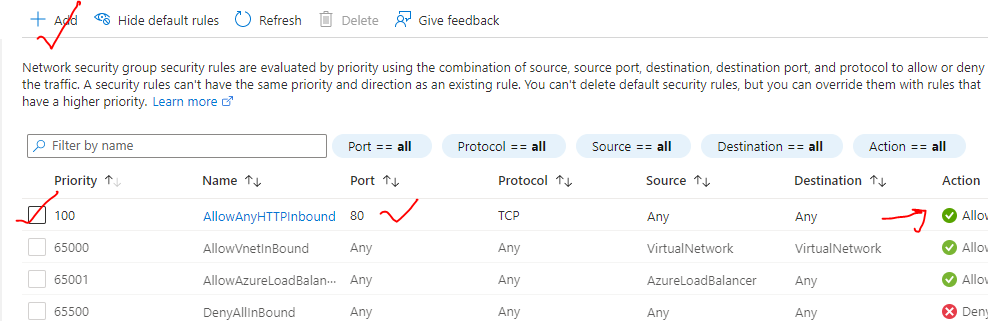


Step 3: RULES ON SUBNET LEVEL

* After the newly created rules is attached to the VM’s subnet. This gets evaluated first
* In the below rules – there is no rule for “80” PORT hence the request for the service running on “80” will be denied



Step 4: ADDING A RULE ON SUBNET LEVEL



## HOW NETWORK SECURITY GROUPS FILTER NETWORK TRAFFIC

* We can use an Azure network security group to filter network traffic to and from Azure resources in an Azure virtual network. A network security group contains [security rules](https://learn.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview#security-rules) that allow or deny inbound network traffic to, or outbound network traffic from, several types of Azure resources. For each rule, we can specify source and destination, port, and protocol.
* We can associate zero, or one, network security group to each virtual network [subnet](https://learn.microsoft.com/en-us/azure/virtual-network/virtual-network-manage-subnet#change-subnet-settings) and [network interface](https://learn.microsoft.com/en-us/azure/virtual-network/virtual-network-network-interface#associate-or-dissociate-a-network-security-group) in a virtual machine. The same network security group can be associated to as many subnets and network interfaces as we choose.

|  |  |
| --- | --- |
| NSG-processing | The following picture illustrates different scenarios for how network security groups might be deployed to allow network traffic to and from the internet over TCP port 80: |

### INBOUND TRAFFIC

**For inbound traffic, Azure processes the rules in a network security group associated to a subnet first, if there's one, and then the rules in a network security group associated to the network interface, if there's one. This includes intra-subnet traffic as well.**

1. **VM1**: The security rules in NSG1 are processed since it's associated to Subnet1 and VM1 is in Subnet1. Unless you've created a rule that allows port 80 inbound, the traffic is denied by the [DenyAllInbound](https://learn.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview" \l "denyallinbound) default security rule, and never evaluated by NSG2, since NSG2 is associated to the network interface. ***If*NSG1*has a security rule that allows port 80, the traffic is then processed by*NSG2*. To allow port 80 to the virtual machine, both*NSG1*and*NSG2*must have a rule that allows port 80 from the internet.***
2. **VM2**: The rules in NSG1 are processed because VM2 is also in Subnet1. Since VM2 doesn't have a network security group associated to its network interface, it receives all traffic allowed through NSG1 or is denied all traffic denied by NSG1. Traffic is either allowed or denied to all resources in the same subnet when a network security group is associated to a subnet.
3. **VM3**: Since there's no network security group associated to Subnet2, traffic is allowed into the subnet and processed by NSG2, because NSG2 is associated to the network interface attached to VM3.
4. **VM4**: Traffic is allowed to VM4, because a network security group isn't associated to Subnet3, or the network interface in the virtual machine. All network traffic is allowed through a subnet and network interface if they don't have a network security group associated to them.

### OUTBOUND TRAFFIC

***For outbound traffic, Azure processes the rules in a network security group associated to a network interface first, if there's one, and then the rules in a network security group associated to the subnet, if there's one. This includes intra-subnet traffic as well.***

1. **VM1**: The security rules in NSG2 are processed. Unless you create a security rule that denies port 80 outbound to the internet, the traffic is allowed by the [AllowInternetOutbound](https://learn.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview" \l "allowinternetoutbound) default security rule in both NSG1 and NSG2. If NSG2 has a security rule that denies port 80, the traffic is denied, and never evaluated by NSG1. ***To deny port 80 from the virtual machine, either, or both network security groups must have a rule that denies port 80 to the internet***.
2. **VM2**: All traffic is sent through the network interface to the subnet, since the network interface attached to VM2 doesn't have a network security group associated to it. The rules in NSG1 are processed.
3. **VM3**: If NSG2 has a security rule that denies port 80, the traffic is denied. If not, the traffic is allowed by the [AllowInternetOutbound](https://learn.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview" \l "allowinternetoutbound) default security rule in NSG2, since a network security group isn't associated to Subnet2.
4. **VM4**: All network traffic is allowed from VM4, because a network security group isn't associated to the network interface attached to the virtual machine, or to Subnet3.

HOW TO DIS- ASSOCIATE NSG FROM VNIC?



|  |  |
| --- | --- |
|  | * Networking 🡪 Network Interface 🡪 Disassociate. This will disassociate the NSG from the VNIC * Once it is disassociated – The VM will be open for all request – as there is no NSG rule associated to it. |

1. ATTACHING AND DETACHING NSG

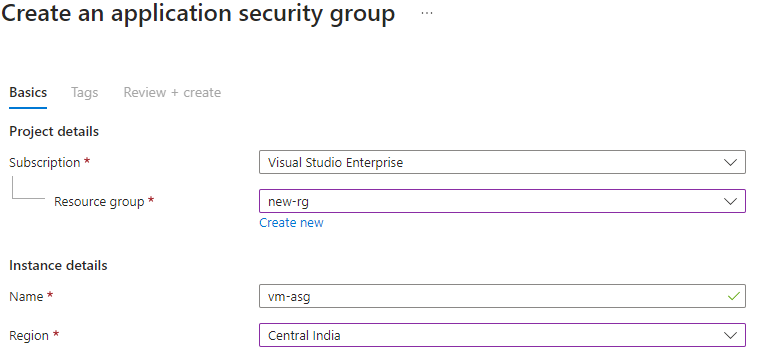
|  |  |  |  |
| --- | --- | --- | --- |
|  | * As the NSG is a separate resource – it can attach and detached form Network interface / Subnet * On the same note, being a separate resource – If we have NSG attached to NIC – we can detach it from NIC and attach the same on Subnet and vice versa * As shown below | | |
|  | | | 1. Step 1 - NSG ATTACHED TO NIC |
|  | | STEP 2: DISASSOCIATE THE NSG FROM NIC | |
|  | | | STEP 3: ATTACH TO THE SAME NSG TO SUBNET |
|  | | | |

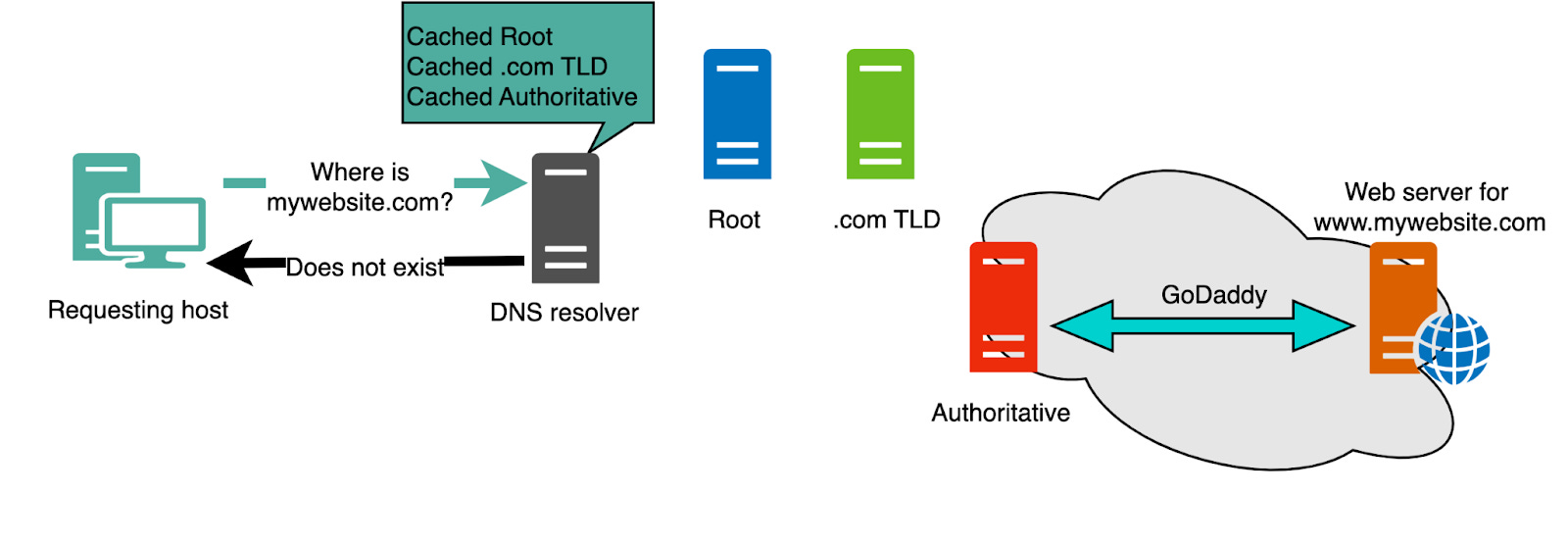
## APPLICATION SECURITY GROUP

|  |  |
| --- | --- |
|  | * **Application Security Groups (ASGs) in Azure are a feature that allows us to group virtual machines (VMs) and define network security policies for those groups.** * ASGs provide a way to simplify network security management by allowing you to define network security rules based on logical groupings of VMs, rather than individual VMs. * ASGs can be used in conjunction with other Azure networking features, such as Network Security Groups (NSGs) and Azure Firewall, to provide a comprehensive network security solution for your Azure infrastructure. |

### SETTING UP APPLICATION SECURITY GROUP

* Application Security Group is a separate resource in Azure – which can be applied on group of VM





<https://www.digwebinterface.com/>

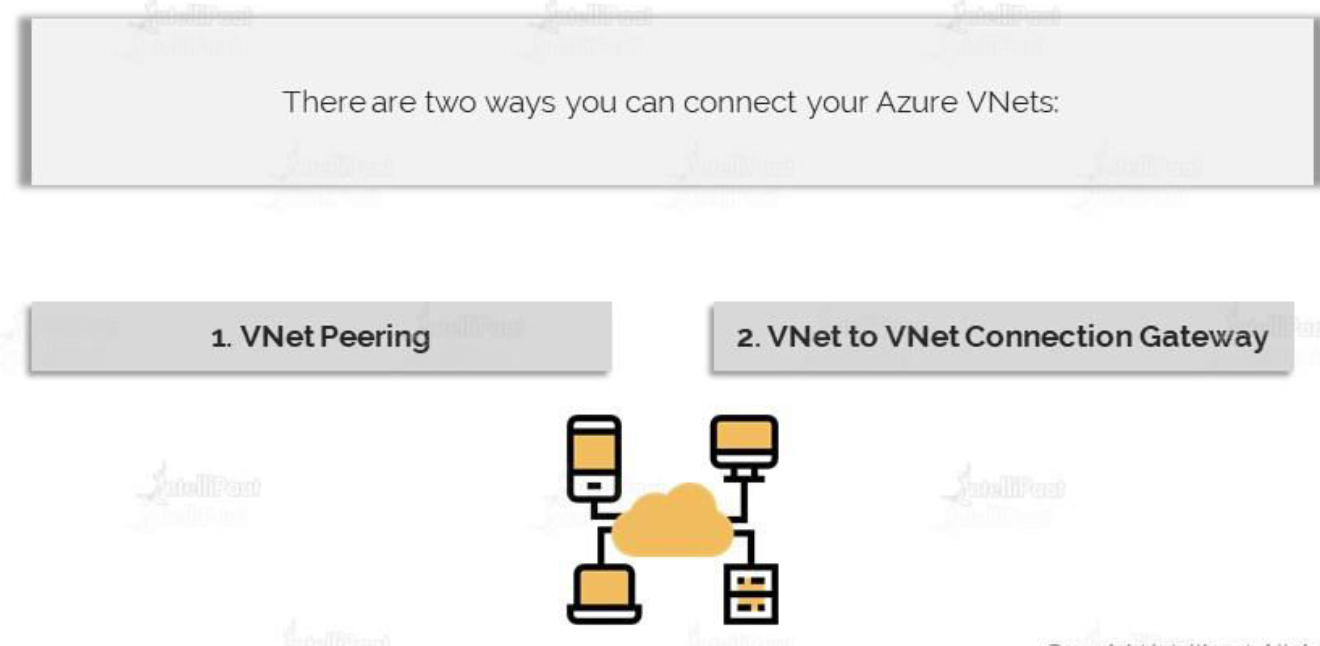
## CONNECTING VIRTUAL NETWORKS

### COMMUNICATION BETWEEN VIRTUAL NETWORKS

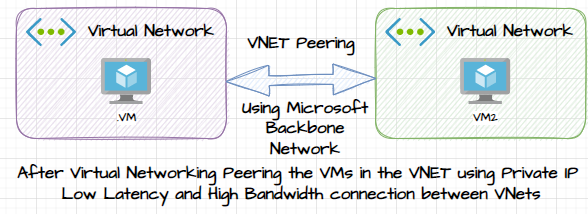
* The virtual network is isolated network in the cloud, hence by default there is no communication can happen between the VNETS.
* The communication between VNETS can be done using ***VNET PEERING*** – where the VNETS can be either with a same region or different region.
* The traffic between virtual machines in peered virtual networks uses the **MICROSOFT BACKBONE INFRASTRUCTURE** not via internet.

**MICROSOFT BACKBONE INFRASTRUCTURE**

* The Microsoft Azure backbone network is a global network infrastructure that connects Azure data centers worldwide. It provides the underlying network connectivity and routing capabilities for all Azure services and resources.
* The Azure backbone network is built on a highly reliable and low-latency infrastructure, allowing for fast and secure communication between Azure services, virtual networks (VNets), and resources deployed in different regions. It ensures that data traverses the network efficiently and reaches its destination with minimal latency.
* ***In a peered VNET – the VMs can communicate using their private IP – even If they belong to a different region. They don’t need any public IP.***
* We can also peer virtual networks that are located across different subscriptions.
* **The virtual networks can't have overlapping CIDR blocks(IP address space cannot overlap)**.



### VIRTUAL NETWORK PEERING



* **Peering connects 2 or more virtual networks in Azure. Traffic between virtual machines in a peered network uses the private Microsoft backbone network and never passes through the public internet.**
* They behave as if the VMs were on the same virtual network.
* **Benefit**s : low-latency, high-bandwidth connection between resources in different virtual networks.

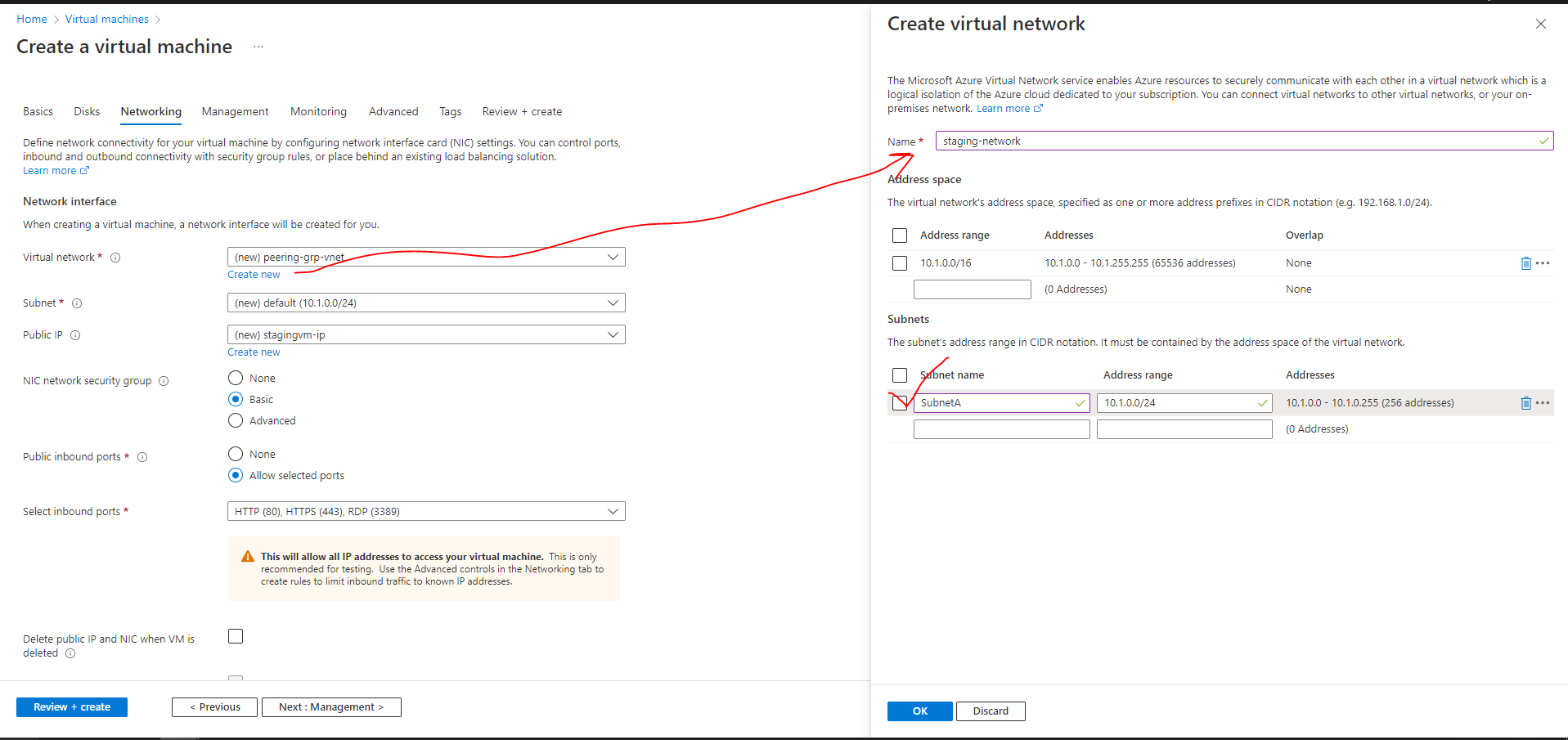
#### TYPES OF PEERING:

* **VIRTUAL NETWORK PEERING**: CONNECTING VIRTUAL NETWORKS WITHIN THE SAME AZURE REGION.
* **GLOBAL VIRTUAL NETWORK PEERING**: CONNECTING VIRTUAL NETWORKS ACROSS AZURE REGIONS.

|  |  |
| --- | --- |
|  |  |
| **VNET PEERING NOT TRANSITIVE** | * If the VNET peering has been done between Region 1 🡪Region 2 and between Region 2🡪 Region 3 . Then doesn’t mean that a VM from Region 1 can able to communicate to a VM in Region 3 * A separate peering has to be done for this communication between Region 1 and Region 3 |

#### SETTING UP VIRTUAL NETWORK PEERING

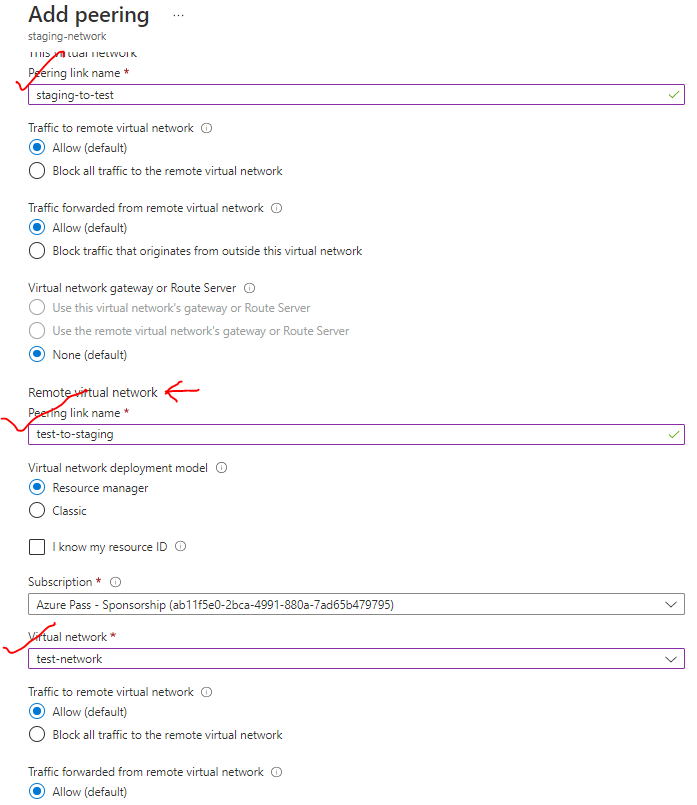
|  |  |
| --- | --- |
|  | * **Within the same VNET – by default the resources on the VNET can communicate using their private IP.** * **Communication using private IP is not possible between 2 different VNets (even if they are in the same region)** * **But- after the VNET Peering – the communication with private IPs will be possible (between the VM deployed in the Vnets).** |



#### PEERING THE VNETS

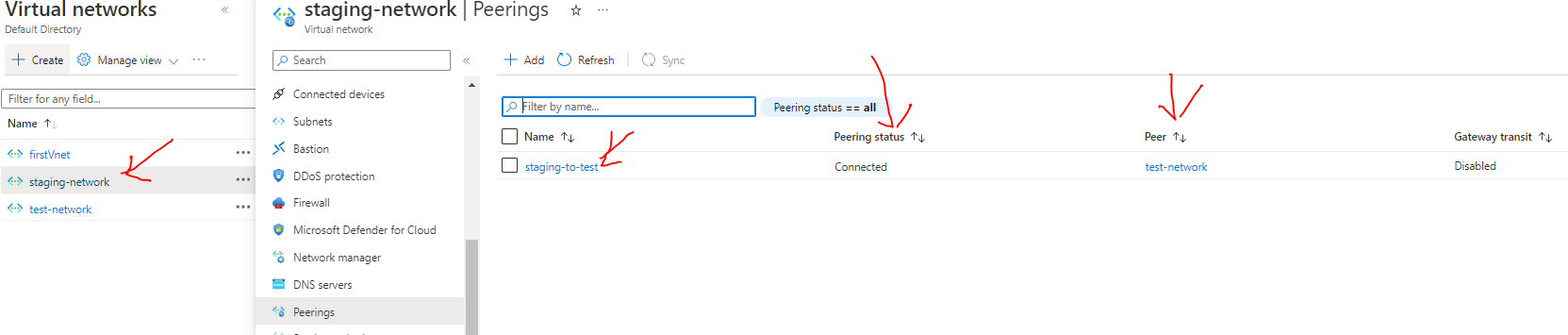
* In peering configuration – we need create the link from both sides. This will allow cross communication between the virtual machines and both networks.
* Go to either of the Vnet 🡪 Peering 🡪 Add (This configuration will establish cross peering between VNets )

|  |  |
| --- | --- |
| Peering link name | We need to provide 2 peering link name one from each VNET |
| Virtual Network | Since we are on Staging network – hence the VNET we want to establish the peering has been provided i.e., “test-network” |

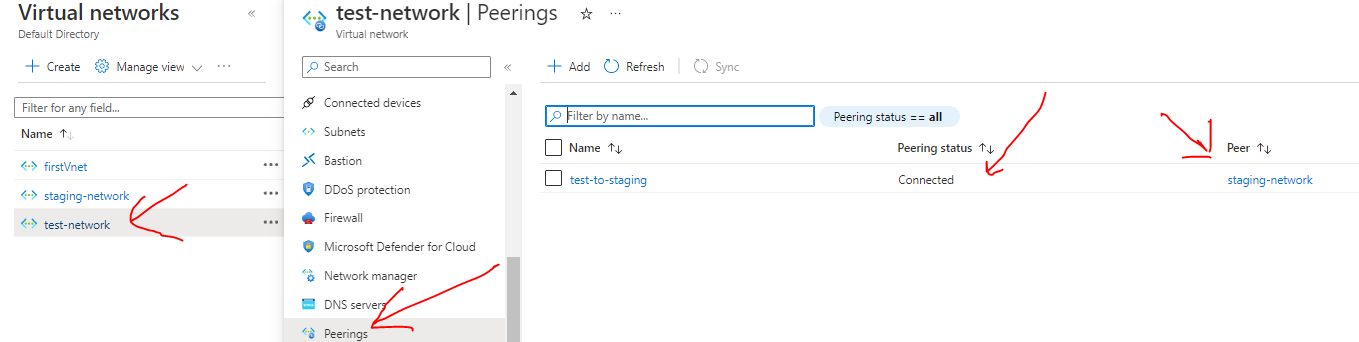


PEERING ESTABLISHED

STAGE TO TEST

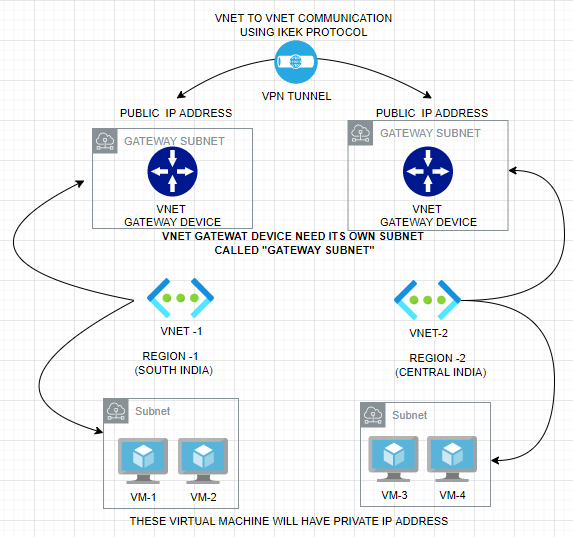


TEST TO STAGE



* **After peering RDP to testvm 🡪 Access the IIS using the private IP of stagevm**

### VNET GATEWAY (VNET TO VNET CONNECTION USING GATEWAY DEVICE)



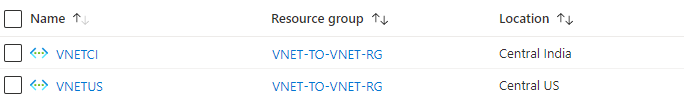
* In VNET-to-VNET connection using gateway device. We need to deploy a VNET gateway device on both VNETs
* The VNET requires, its own a dedicated Subnet – which we refer as **Gateway Subnet**
* When we deploy a VNET gateway device it will have a Public IP address (static IP address) associated with it, by default
* Between these Virtual Gateway device – we establish a VPN Tunnel. The protocol used for this communication is **IKEK**
* Now the VMs in VNET-1 and VNET-2 can be able to communicate with each other via VPN connection(tunnel) using their private IP address.

#### SETTING UP GATEWAY COMMUNICATION

1. STEP 1: Create 2 V-NETS -Let’s say in two regions (Central India and Central US). While creating the VNET – a default subnet is also created where we will deploy the VMs.
2. STEP 2: While creating the V-NET make sure we select different IP address range for both the VNETs

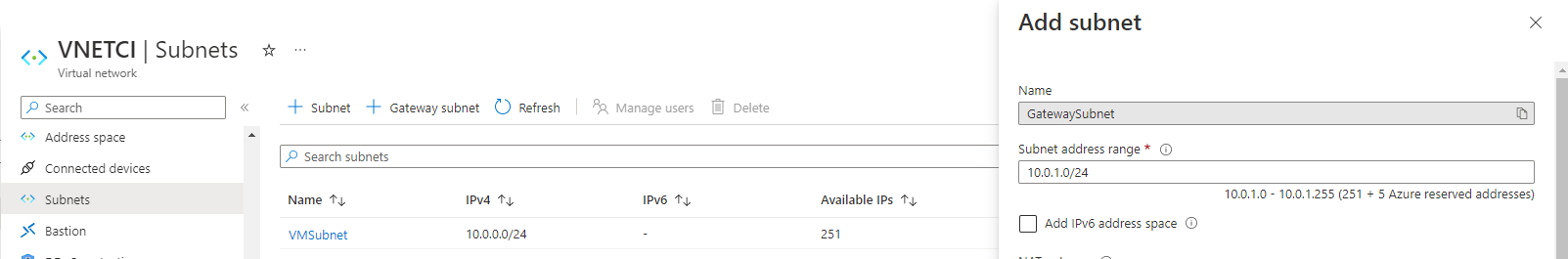
|  |  |
| --- | --- |
| **VNET -1**  **(CENTRAL INDIA)** |  |
| **VNET -2**  **(CENTRAL US)** |  |

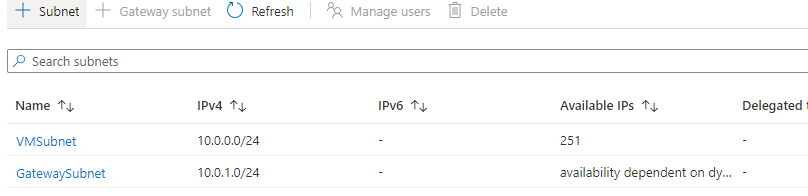
1. STEP 3: VNET CREATED



1. STEP 4: DEPLOYING A GATEWAY SUBNET

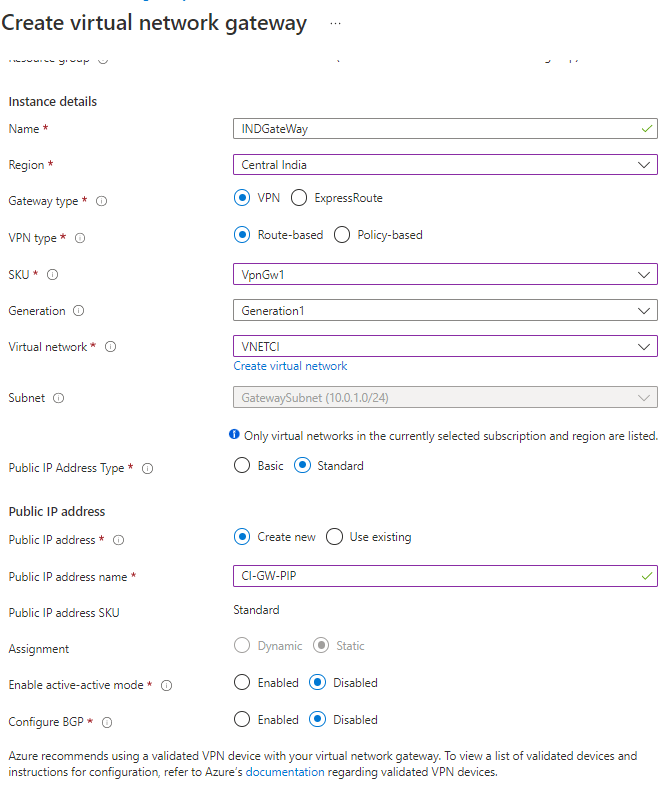
* To deploy the Gateway subnet 🡪 Navigate to Corresponding subnet

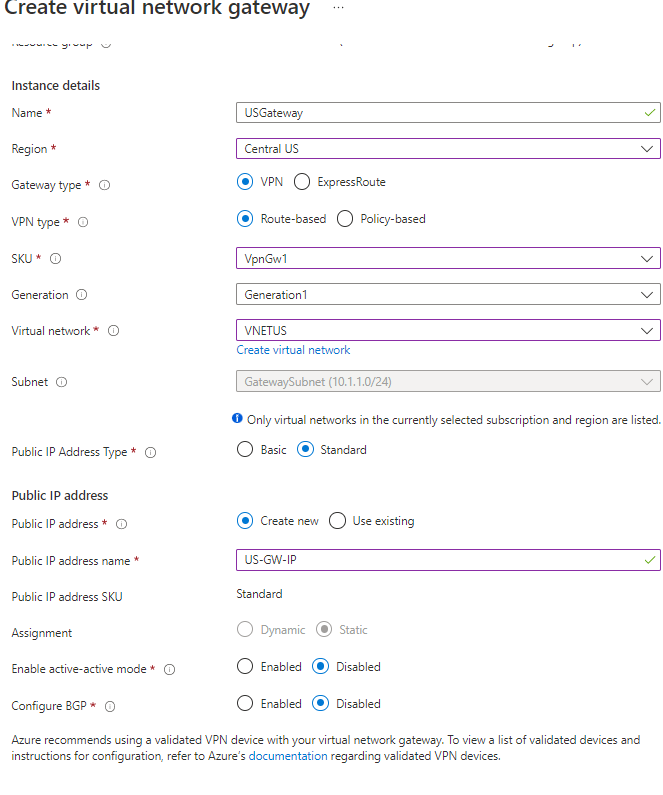




1. STEP 5: DEPLOYING VITUAL NETWORK GATEWAY DEVICE

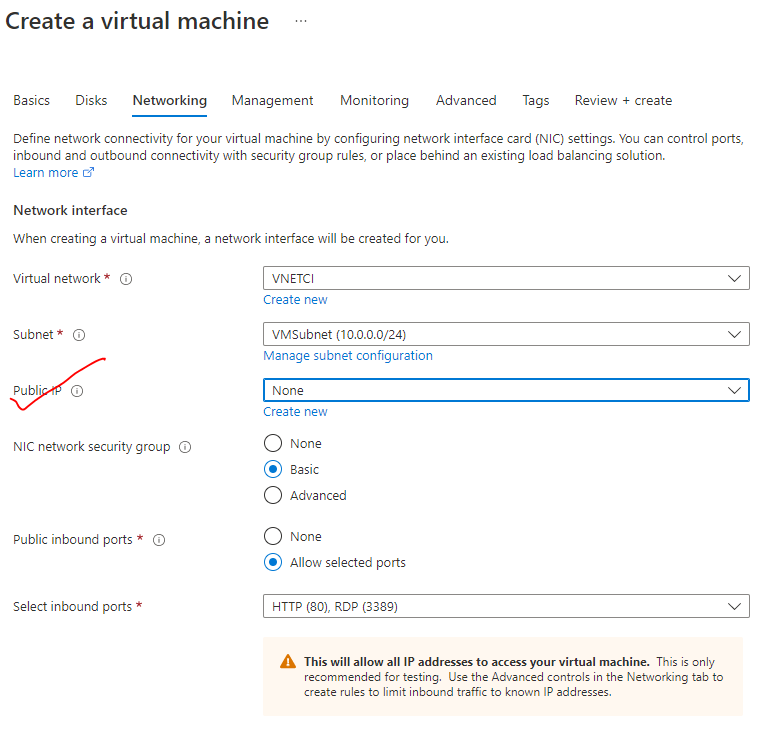
We need to deploy gateway device in both the Virtual networks. As shown below



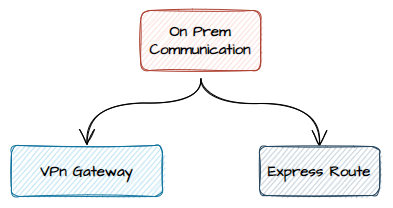


1. STEP 6: DEPLOYING VM IN THE VNET

* ***We create VM in a subnet – as the communication will happen using private IP -hence we don’t need have public address***



## CONNECTING TO - ON PREMISE



### VPN GATEWAY

|  |  |
| --- | --- |
| How Does a VPN Work Illustration | * VPN stands for Virtual Private Network. It is a technology that allows for the **creation of a secure and encrypted connection over a public network**, typically the internet. * A VPN provides a secure tunnel for data transmission between a user's device and a remote server or network, ensuring the privacy and integrity of the data being transmitted. |

* **WHEN WE WANT TO SECURELY COMMUNICATE BETWEEN YOUR AZURE RESOURCES AND THE ON-PREMISES NETWORK – WE MAKE USE OF VPN GATEWAY.**
* *Let's start with a virtual network gateway. A virtual network gateway is composed of two or more virtual machines that are deployed to a specific subnet which is called the gateway subnet. The VMs that are located in the gateway subnet*

*are created when we create the virtual network gateway.*

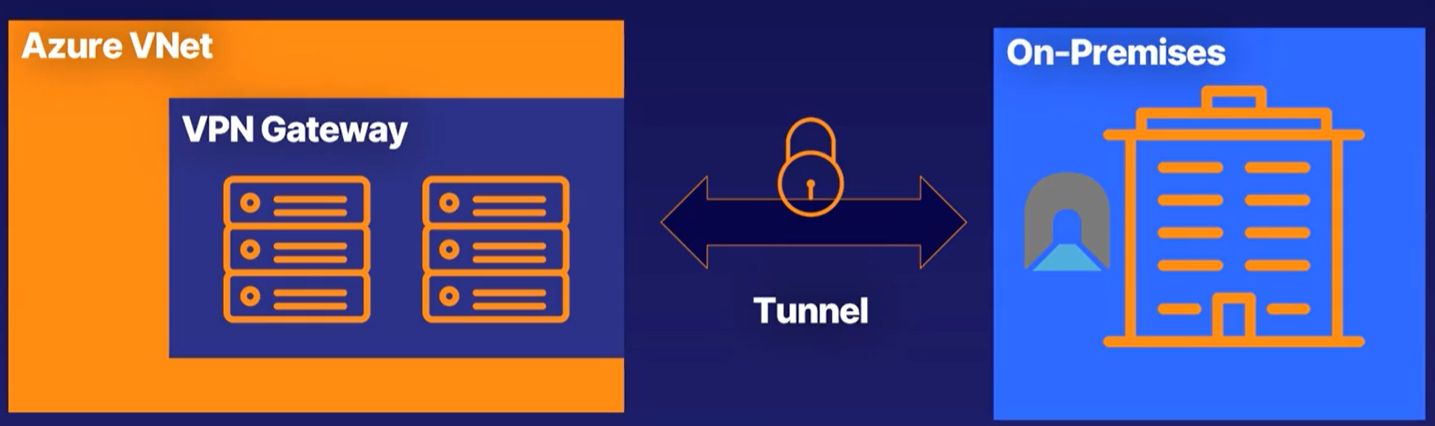
* **A VPN gateway is a specific type of virtual network gateway that is used to send encrypted traffic between an Azure virtual network and an on-premises location over the public internet.**



* Example - Let's say we a company that has its own on-premises infrastructure, wanting to some resources to the cloud to i.e. a hybrid solution- with some of the data on-premises and some on Azure. **To communicate securely between the two, we create a VPN gateway, which is a specific kind of VNet gateway.**

**A VPN GATEWAY IS A KEY PART OF HAVING A SECURE AND AVAILABLE HYBRID CLOUD ARCHITECTURE.**

COMPONENTS OF VPN GATEWAY

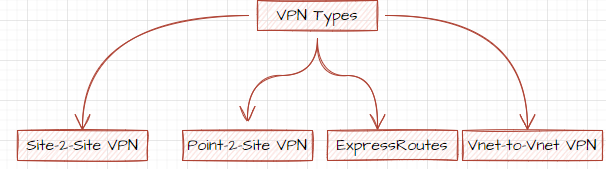


**There are three components a VPN gateway**.

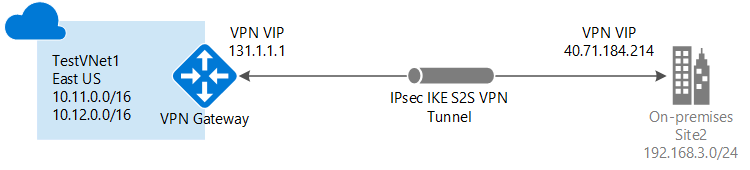
1. *An Azure VNet with a VPN gateway attached having its own public IP address -*
2. *A secure connection called a tunnel, which has one of a number of encryption mechanisms(****IPSec****),*
3. *And on-premises network with a complementary gateway that can accept the encrypted data.*

#### TYPES OF VPN CONNECTION

In Azure, there are several types of VPN connections that we can utilize to **establish secure connections between on-premises networks and Azure virtual networks**. Here are the commonly used types of VPN connections in Azure:

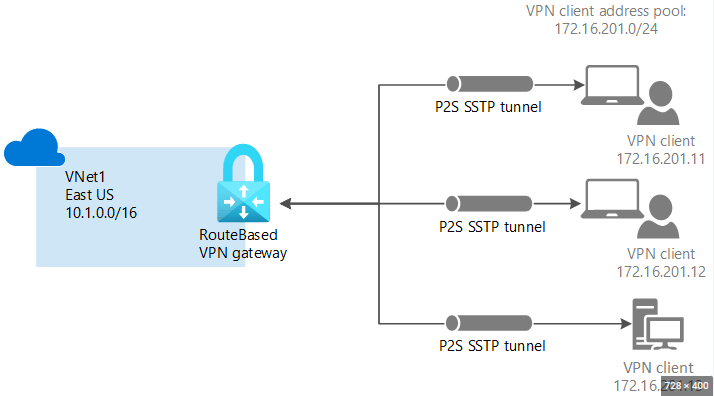


##### SITE-TO-SITE VPN



* Azure Site-to-Site VPN allows us to establish a secure connection between entire on-premises network and an Azure virtual network. It enables us to extend our on-premises network to Azure, allowing bidirectional communication between the two environments.
* As it’s a VPN connection, it’s is a private communication (using private IPs of the resources) between the sites via a virtual tunnel using IKE protocol.

##### POINT-TO-SITE VPN



* **Azure Point-to-Site VPN enables individual client devices, such as laptops or desktops, to securely connect to resources in the Azure virtual network using the private IP address.**
* It provides remote access to Azure resources from anywhere, allowing users to connect to the virtual network securely over the internet. This type of VPN connection is useful for remote workers or scenarios where we need secure access to Azure resources from specific client devices.
* For P2S VPN connectivity we need to provision Azure Virtual Network Gateway Device
* A secure connection is first made from the client machine onto a virtual network. And then via that VPN connection, we can then use the private IP address of the Azure virtual machine for the communication. Hence we don’t need a public IP address of the VM

|  |  |
| --- | --- |
| **POINT TO SITE CONNECTION** | **SITE TO SITE CONNECTION** |
| 1. P2S connection can be used when small number of clinet machines has to connect with the VNET. | 1. S2S connections are used when we want to connect an entire on premises network onto as the virtual network. |

### EXPRESS ROUTE

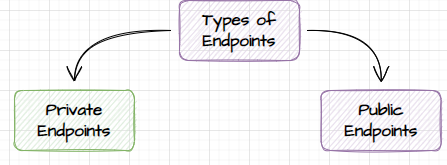


* ExpressRoute is a service provided by Microsoft Azure that **allows us to establish a private and dedicated network connection between the on-premises network and Azure.**
* It provides a more reliable, faster, and more secure connection compared to traditional internet-based connections.
* **Example – If the company need a data which are on-prem and on Azure which has to be highly available and periodically migrated – We should prefer express route connection.**

KEY POINTS

* PRIVATE CONNECTION:
  + **ExpressRoute enables us to establish a private and dedicated connection to Azure, bypassing the public internet.**
  + **This ensures better security, lower latency, and higher bandwidth for the data transfer.**
* SERVICE PROVIDERS:
  + ExpressRoute can be provisioned through various service providers globally.
  + These service providers offer connectivity options, such as MPLS or Ethernet, to connect the on-premises network to Azure.
* DEDICATED CIRCUITS:
  + ExpressRoute circuits provide the physical connectivity between the on-premises network and Azure.
  + They can be provisioned with different bandwidth options, ranging from 50 Mbps up to 100 Gbps, depending on our requirements.
* HYBRID SCENARIOS
  + **ExpressRoute is commonly used for hybrid cloud scenarios, where we want to extend the on-premises network to Azure and have seamless connectivity between your resources.**

## TYPES OF ENDPOINTS

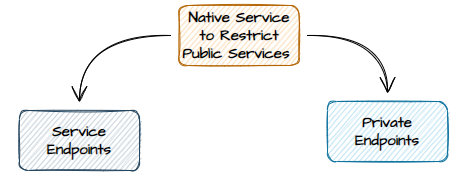


### PUBLIC ENDPOINT

* By default the PaaS service like Azure Storage, Azure SQL etc. These services are publicly reachable or or publicly exposed over the public internet.

|  |  |
| --- | --- |
| * When accessing these public services over a VNET (specifically from VM in the VNET) - **traffic from that virtual network to these managed services (like Azure Storage) traverses the public internet**. * **Additionally, these managed services end points are by default public endpoints, which are exposed to the public**.   *Note – Public endpoint - does not mean that the contents of those resources are accessible to the public. We still need proper authentication to grant access. However, these resources are still publicly exposed or publicly reachable,*  *which in scenarios in which your services contain sensitive content, this could be a little bit of a problem.* |  |

To limit the public exposure we have 2 native services and have private communication within Azure

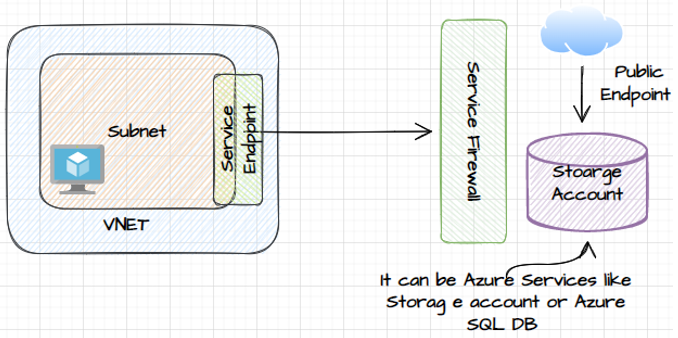


### SERVICE ENDPOINT

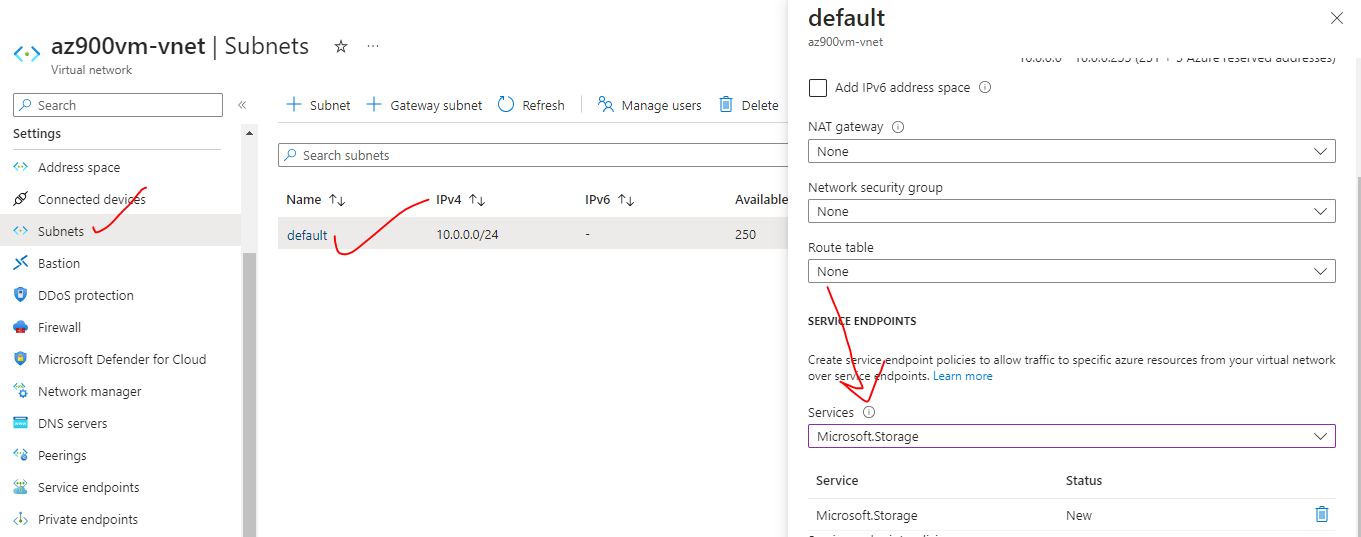
* Service endpoints allow **secure connectivity from Azure VMs that are part of Azure virtual network** to PaaS services such as Azure storage accounts or Azure Web apps, etc.
* There is no cost associated with service endpoint

EXAMPLE

* Let's say we have an Azure VM that is part of an Azure virtual network, that doesn't have a public IP address. The VM has an application that needs to connect onto an Azure storage account. Hence for private communication between them

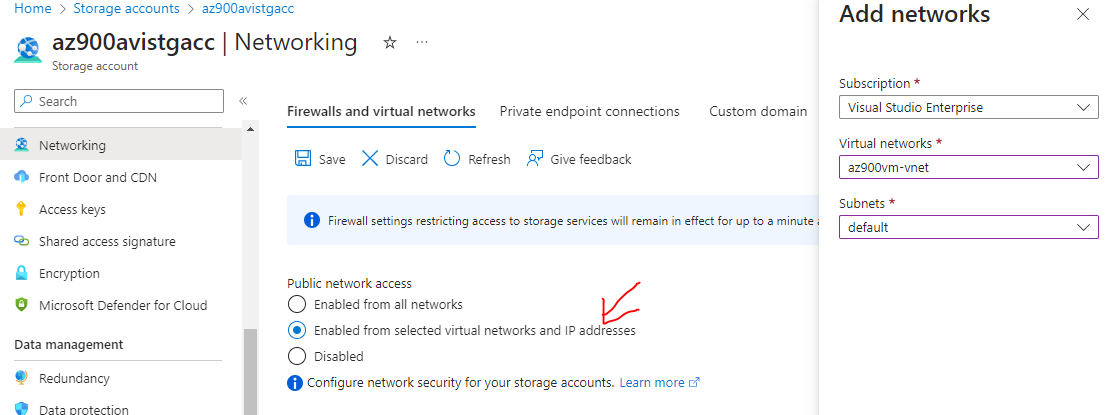


#### STEP 1: FROM VNET WE NEED TO ENABLE SERVICE ENDPOINT FOR STORAGE SERVICE



The Service endpoint has been created for the subnet of which the VM is part of.

#### STEP 2: FIREWALL SETTING IN STORAGE ACCOUNT

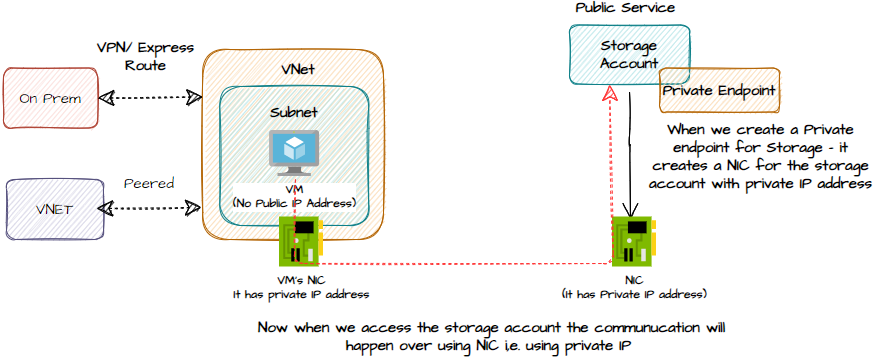




#### LIMITATIONS OF SERVICE ENDPOINTS

* Service Endpoints provides secure access to only Azure virtual networks. For non-Azure services or on-premises locations - the service endpoint provides private access over the public internet. However, we restrict to a small portion of public IPs only even though it is still going over the public internet. In other words, that connection is still not truly private, and there is still some public internet exposure.
* Service endpoints provide private access to the entirety of a managed service not a specific instance of a managed service. For example, when we set up service endpoints with a VNet subnet, it provides private access to the entirety of the Azure Storage service, not just to a single individual storage account.

### PRIVATE ENDPOINT



* The traffic between the virtual network and the PasS services goes via the Microsoft Backbone Network.

EXAMPLE

* *Let's say we have a storage account, which is a public service available on the Internet. When we implement private endpoint, it creates a Network interface card with a private IP*
* *When we access the storage account - the communication between this VM and the storage account will be done via this private IP address.*
* *Along with private IP – it also creates Azure private DNS zone. The private IP address of the VNIC mapped to the private DNS.*
* *The communication with the services like storage account can happen using private IP, (i.e., using DNS)*

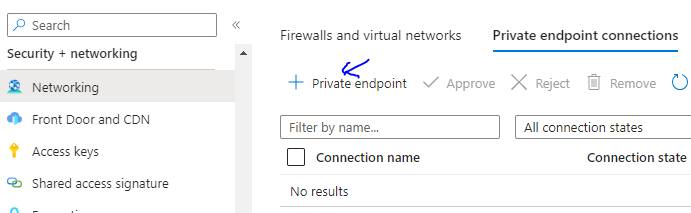
#### COMMUNICATION USING PRIVATE ENDPOINT

* *The Peered VNET can also do the communication with storage account using private endpoint.*
* *Similarly – from On-Prem environment which are connected using Site 2 Site VPN or Express route to the VNET/Subnet can also connect to storage account using Private endpoint.*
* *To validate the connectivity to a storage using private endpoint:* [*https://github.com/MicrosoftDocs/azure-docs/blob/main/articles/private-link/tutorial-private-endpoint-storage-portal.md*](https://github.com/MicrosoftDocs/azure-docs/blob/main/articles/private-link/tutorial-private-endpoint-storage-portal.md)

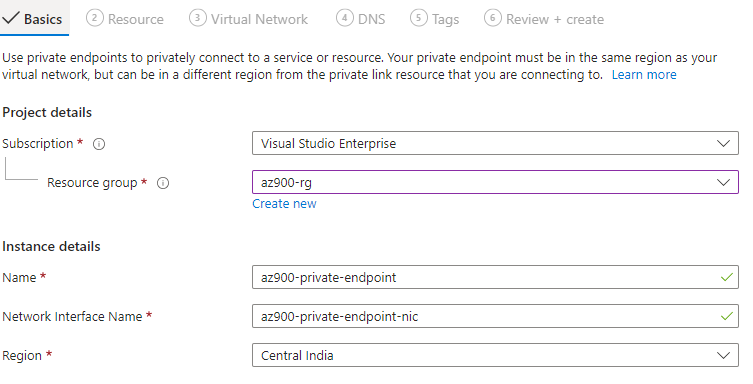
#### SETTING UP PRIVATE ENDPOINT (FOS STORAGE ACCOUNT)

*Step 1: Create a Storage Account*

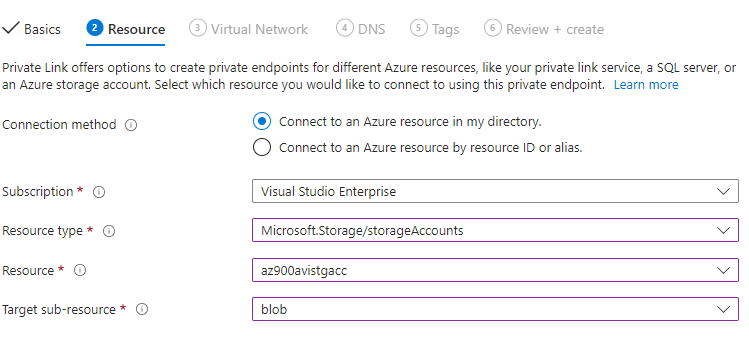
*Step 2: Set up private endpoint for Storage Account*

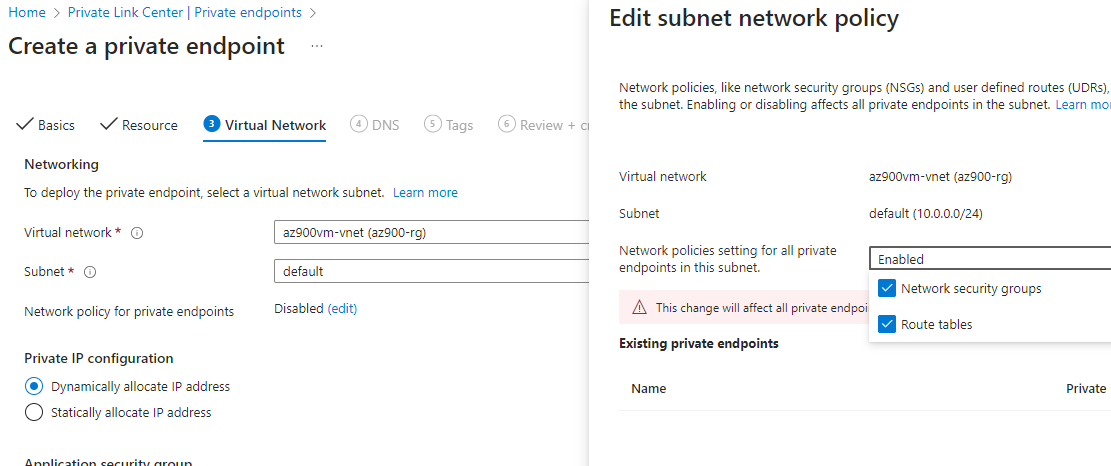


* The private endpoint a separate resource in Azure.
* The below wizard creates a Network interface Card

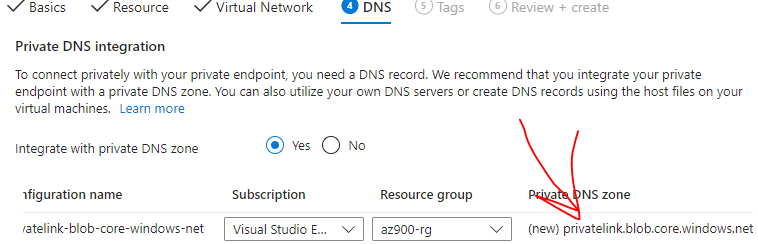


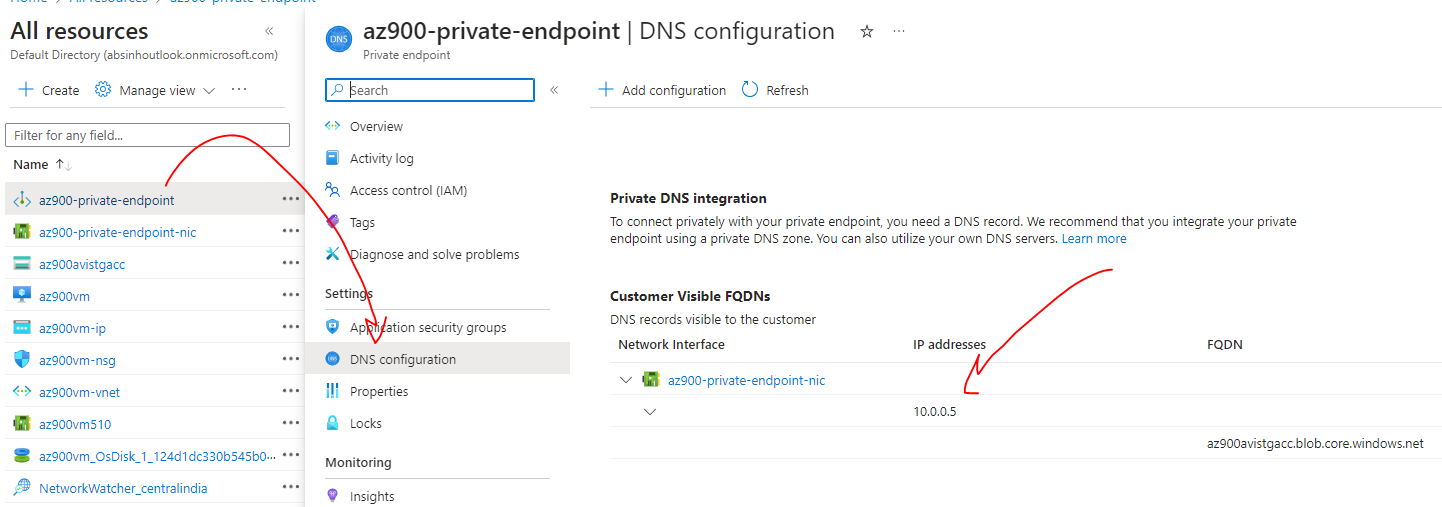
*Step 2: Select the Resource to which the endpoint will connect. Below private endpoint will target the ”blob” service*



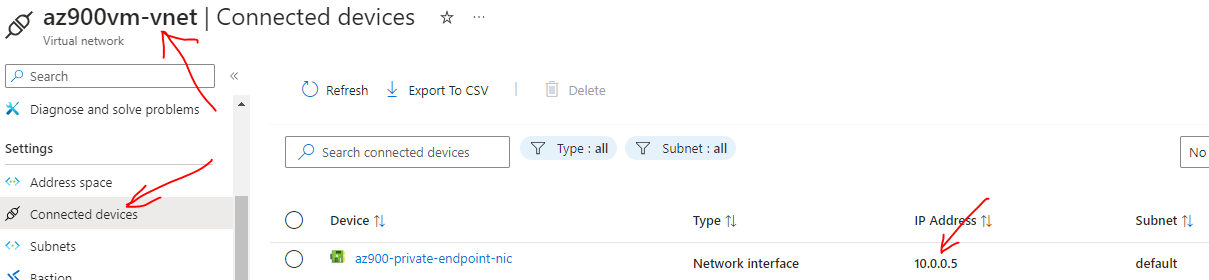


*Step 3: Azure provides the DNS name to the private endpoint(Private IP address)*

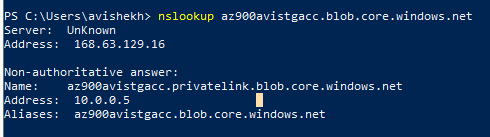




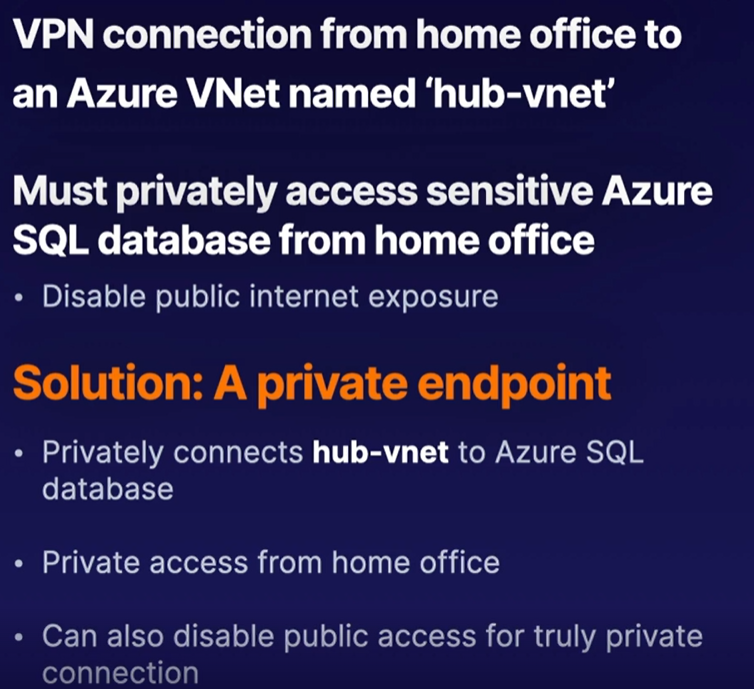
*View from VNET end (Connected devices will show the NIC attached to the private endpoint)*

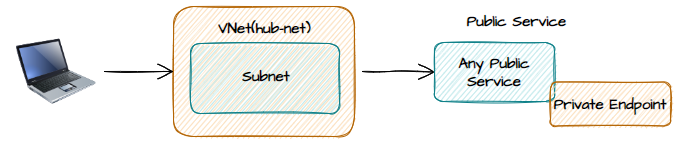


*To test the connection from the VM in the VNET*

**

#### USE CASE



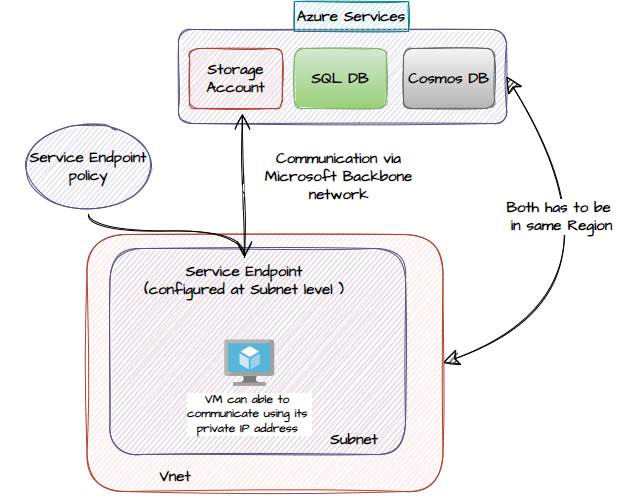


## SERVICE ENDPOINTS AND SERVICE ENDPOINT POLICIES

* Service Endpoints and Service Endpoint Policies are features that allow you to secure and optimize network traffic between your Azure Virtual Network (VNet) and specific Azure services.

SERVICE ENDPOINTS

* Service Endpoints enable us to extend the VNet's private address space and allow resources within the VNet to connect to specific Azure services directly over the Azure backbone network.
* By using Service Endpoints, the traffic between your VNet and the Azure service remains within the Microsoft Azure network, providing a more secure and optimized connection compared to going through the public internet.
* Service Endpoints are associated with specific Azure services (e.g., Azure Storage, Azure SQL Database, Azure Cosmos DB, etc.) and require us to create a Service Endpoint connection between VNet and the Azure service.

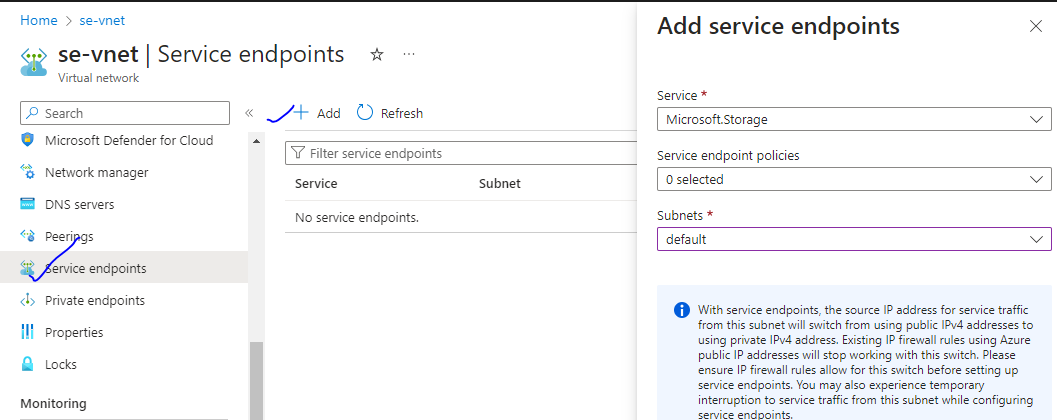


SERVICE ENDPOINT POLICIES:

* Service Endpoint Policies allow you to further control and secure network traffic to Azure services by defining network access rules for specific subnets within your VNet.
* With Service Endpoint Policies, you can restrict access to specific Azure services only from selected subnets within your VNet, providing granular control over network traffic.
* You can define inbound and outbound network access rules in the Service Endpoint Policies to allow or deny traffic to and from the Azure service endpoints.
* Service Endpoint Policies work in conjunction with Network Security Groups (NSGs) to provide additional layers of network security.

### CONFIGURING SERVICE ENDPOINT

ADD SERVICE ENDPOINT IN THE SUBNET



ENABLE THE SUBNET

