Microsoft Azure Administration

Module-04: Azure Networking Services

Contents

- Networking concepts Primer
- Overview of Azure Virtual Network
- Subnetting
- Network Security Groups (NSG)
- Azure Load Balancer
- Azure Application Gateway
- Azure Traffic Manager
- Virtual Network connectivity options
- Azure Virtual Network: Pricing
- Azure Network Watcher
- Azure Bastion Service

Networking concept Primer

- IP Address: 10.0.1.10 – IPv4 | Public and Private
- CIDR (/): 10.0.1.10 /24 = 256 IP Address | 50 IP Address
- Network and Subnet
- Router
- NAT Instance
- Bastion Host / Jump Server

IP Addressing and CIDR

IP Address and CIDR notations (/)

- CIDR stands for **Classless Inter-domain Routing**
- CIDR is a method for allocating IP addresses and for IP routing which improves the allocation of IP addresses
- CIDR notation (slash or '/') is a way of representation of IP addresses, in which an address or routing prefix is written with a suffix indicating the number of bits of the prefix, such as 192.0.2.0/24 for IPv4

Understanding CIDR

- A CIDR has two components:
  1. The **Base IP (XX.XX.XX.XX):** The base IP represents an IP contained in the range
  2. **The Subnet Mask (e.g., /24):** The subnet masks defines how many bits can change in the IP

- The subnet mask can be represented in below two forms:
  1. 255.255.255.0 less common
  2. /24 more common (used in Azure/AWS services)

Understanding CIDR

Digit after slash indicates the number of Network bits (e.g., /24)

Number of Host bits = 32 – No. of Network bits

Number of Total host IP in a CIDR = $2^{\text{Number of Host bits}}$

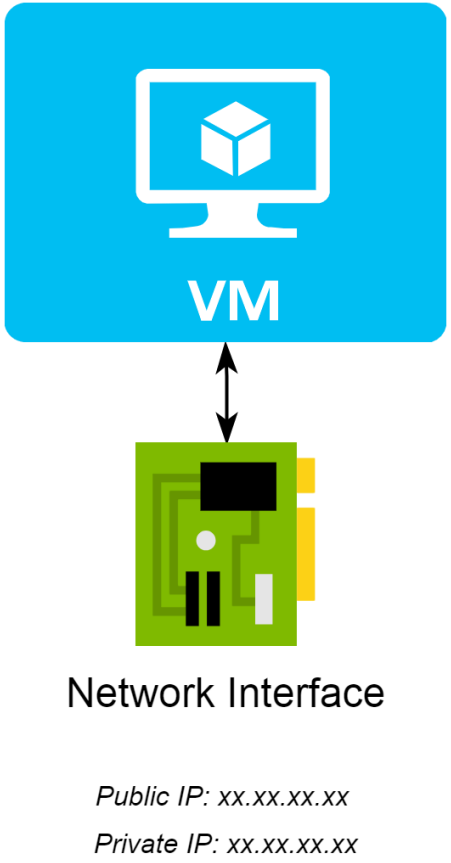
Understanding CIDR – Some Exercise

1) 192.168.10.0 /28 = ? IP Addresses
2) 10.20.30.0 /24 = ? IP Addresses
3) 10.10.0.0 /16 = ? IP Addresses
4) 20.30.10.50 /32 = ? IP Address
5) 0.0.0.0/0 = ? IP Addresses
6) 10.20.30.0/26 = ? IP Address
7) 10.10.1.0/27 = ? IP Address

The Network Interfaces

Overview of Network Interface

- A network interface enables an Azure Virtual Machine to communicate with internet, Azure, and on-premises resources
- A NIC is a component which holds the Public IP and the private IP of the VM
- You can associate the Network Security Group to the NIC

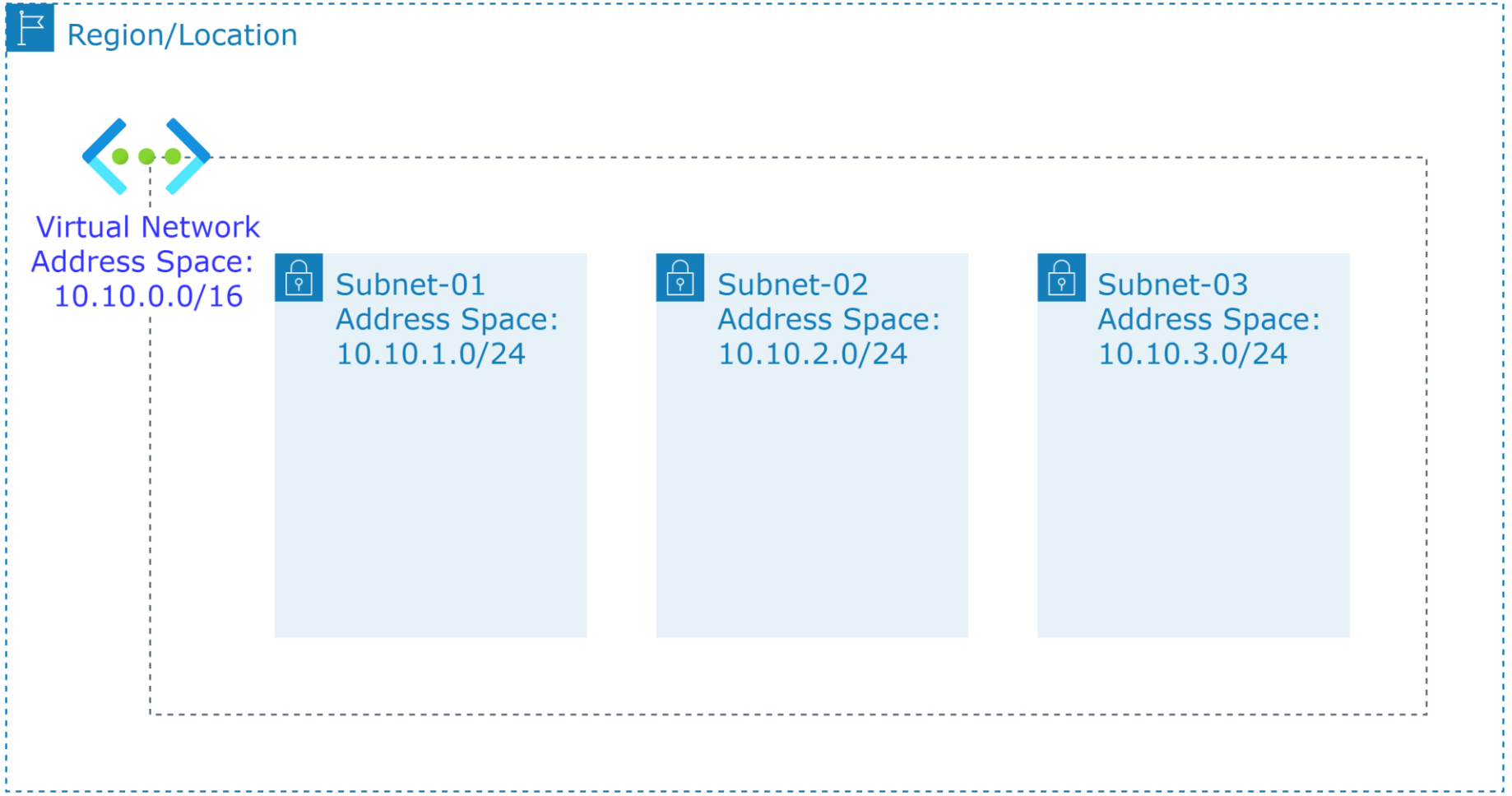


Azure Virtual Networks

Overview of Azure Virtual Network (VNet)

- Azure Virtual Network (VNet) is the fundamental building block for your private network in Azure
- VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks
- Azure virtual network enables Azure resources to securely communicate with each other, the internet, and on-premises networks

Overview of Azure Virtual Network (VNet)



Azure Virtual Network: Common scenarios

Key scenarios that you can accomplish with a virtual network include:

- Communication of Azure resources with the internet
- Communication between Azure resources
- Communication with on-premises resources
- Filtering network traffic
- Routing network traffic
- Integration with Azure services

Azure Virtual Network: Concepts

- **Address space**: When creating a VNet, you must specify a custom private IP address space using public and private (RFC 1918) addresses. Azure assigns resources in a virtual network a private IP address from the address space that you assign. E.g., 10.0.0.0/16, the VM will be assigned a private IP like 10.0.0.4.
- **Subnets**: Subnets enable you to segment the virtual network into one or more sub-networks and allocate a portion of the virtual network's address space to each subnet
- **Regions**: VNet is scoped to a single region/location; however, multiple virtual networks from different regions can be connected together using Virtual Network Peering
- **Subscription**: VNet is scoped to a subscription. You can implement multiple virtual networks within each Azure subscription and Azure region

Azure Virtual Network – Key Points

- A **virtual network** (VNet) allows you to specify an IP address range for the VNet, add subnets, associate network security groups, and configure route tables
- A subnet is a range of IP addresses in your VNet. You can launch Azure resources into a specified subnet
- Use a public subnet for resources that need to connect to the Internet and a private subnet for resources that won't be connected to the Internet
- To protect the Azure resources in each subnet, use network security groups

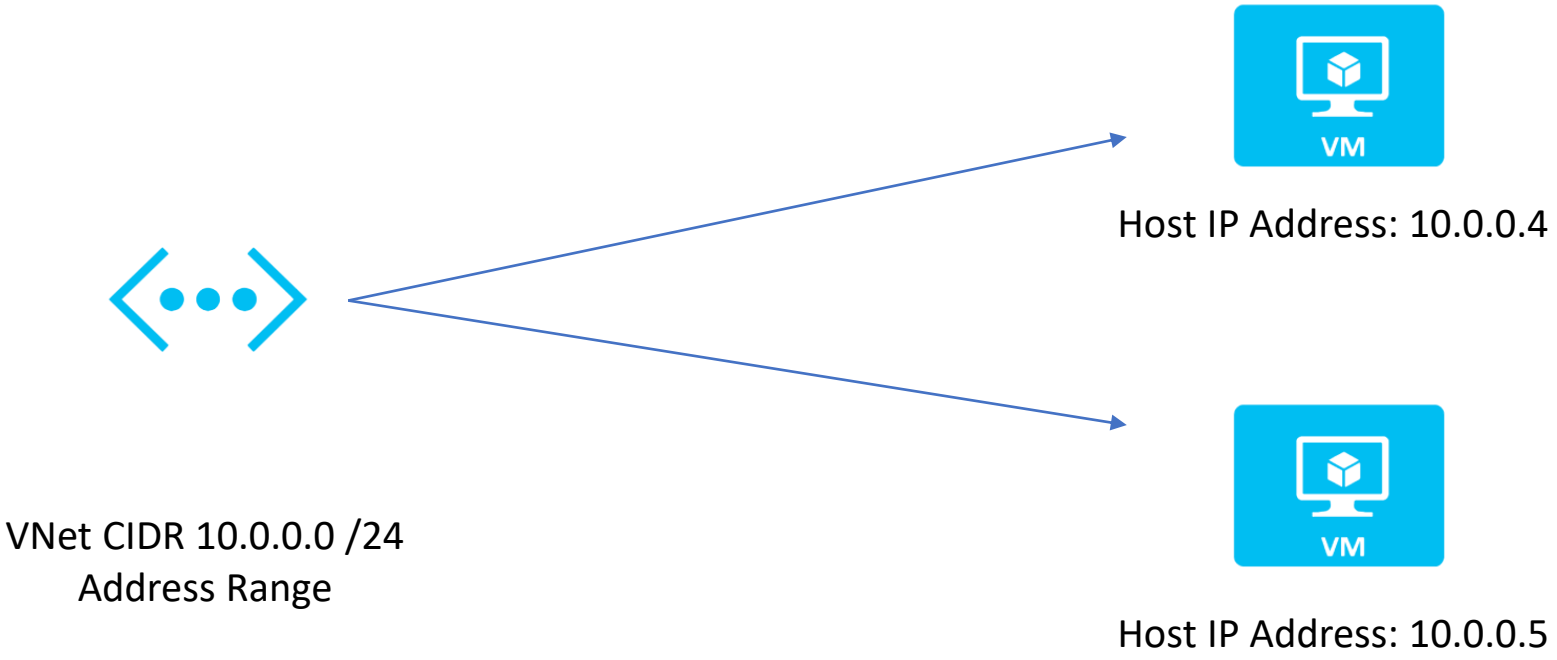
Azure Virtual Network – Use case

1. VNet with a single public subnet
2. VNet with **public** and **private** subnets (NAT)

Quick Note – Address Spaces

IP Addressing

Virtual Network CIDR – 10.0.0.4 /24



What address ranges can I use in my VNets?

- You can have multiple VNet in a Region
  - Smallest supported subnet is /29 = 8 IP Addresses
  - Largest support subnet is /2 = 2^{30} IP (recommended is /16 = 2^{16} IP Addresses)
- VNet is private, only the Private IP ranges are allowed, i.e. :
  1. 10.0.0.0 - 10.255.255.255 (10/8 prefix)
  2. 172.16.0.0 - 172.31.255.255 (172.16/12 prefix)
  3. 192.168.0.0 - 192.168.255.255 (192.168/16 prefix)

Note: Other address spaces may work but may have undesirable side effects

Ref: https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-faq

Restrictions on using IP addresses within subnets

- Azure **reserves 5 IP addresses** within each subnet
- These are x.x.x.0-x.x.x.3 and the last address of the subnet. x.x.x.1-x.x.x.3 is reserved in each subnet for Azure services
  - **x.x.x.0**: Network address
  - **x.x.x.1**: Reserved by Azure for the default gateway
  - **x.x.x.2, x.x.x.3**: Reserved by Azure to map the Azure DNS IPs to the VNet space
  - **x.x.x.255**: Network broadcast address for subnets of size /25 and larger. This will be a different address in smaller subnets

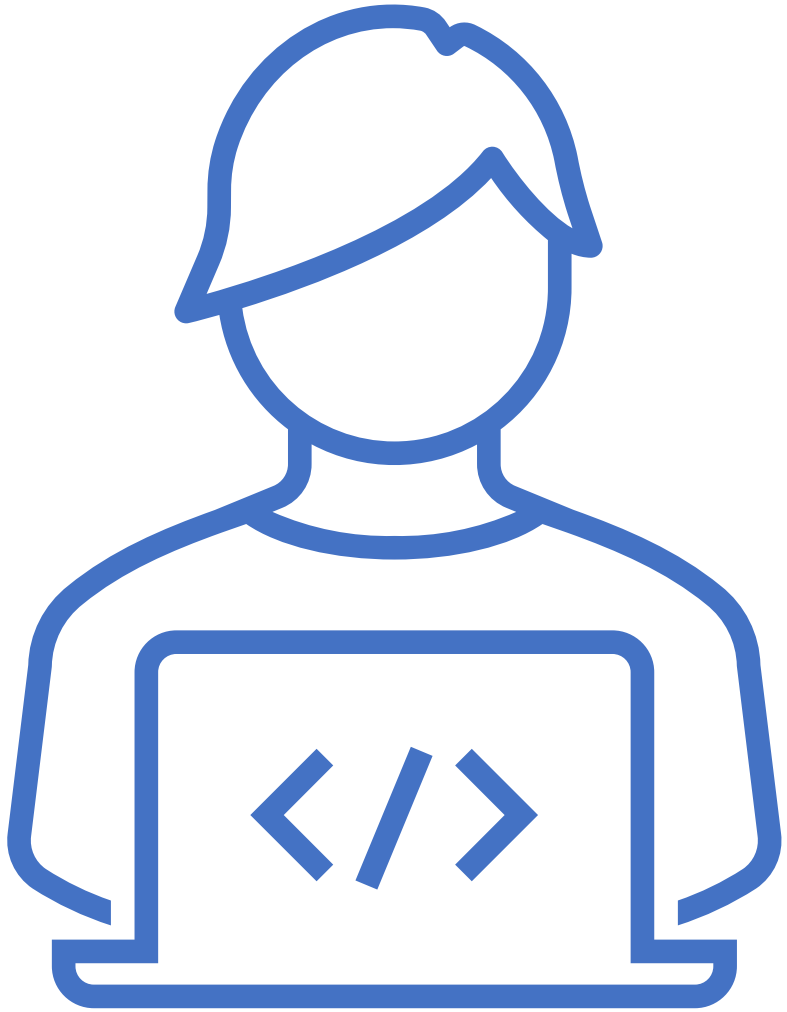
Ref: https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-faq

Azure Virtual Network: Important Points

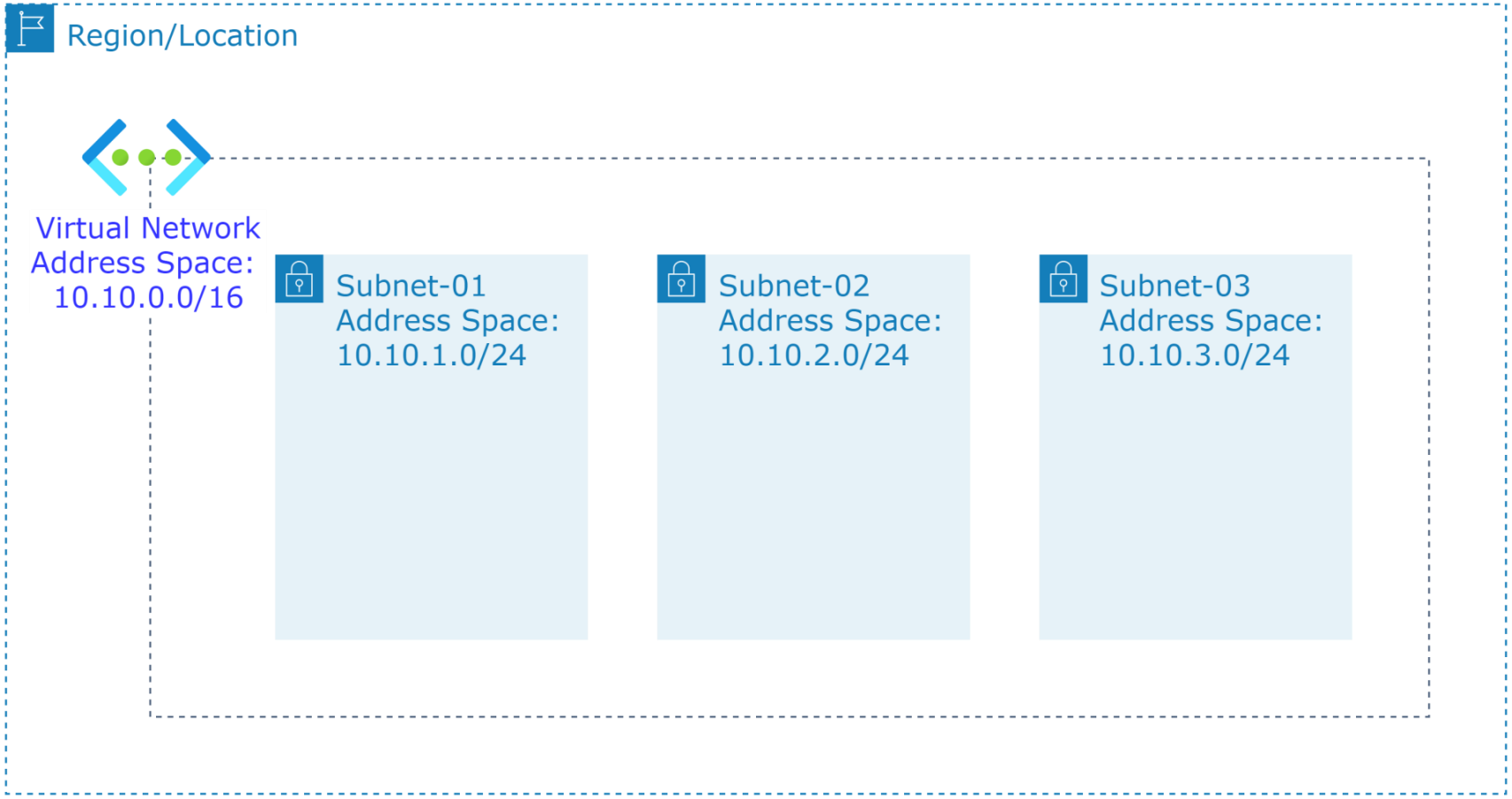
- When you create a VNet, you must specify a range of IPv4 addresses for the VNet in the form of a CIDR block (example: 10.0.0.0/16)
- A CIDR block must not overlap with any existing CIDR block that's associated with your VNet
- You can add multiple subnets in each Availability Zone of your VNet's region
- The CIDR block size of an IPv4 address is between a /16 netmask (65,536 IP addresses) and /29 netmask (8 IP addresses)

Note: The 5 reserved addresses in each CIDR block is not available for you to use and cannot be assigned to any virtual machines

Hands-on Labs

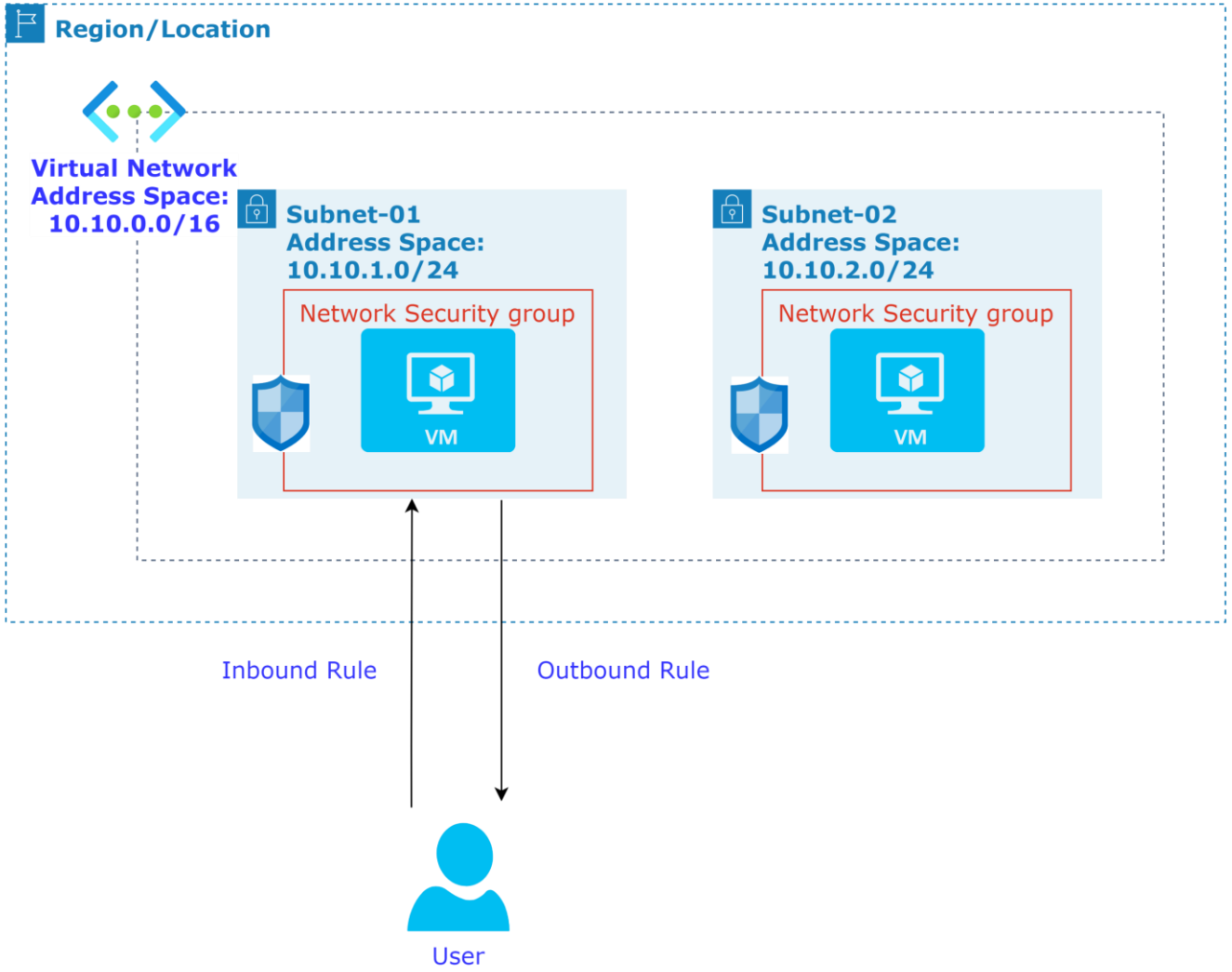


Lab: Create an Azure Virtual Network and Subnets



Network Security Groups (NSG)

Understanding Network Security Groups



Understanding Network Security Groups

- You 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 that allow or deny inbound network traffic to, or outbound network traffic from, several types of Azure resources
- For each rule, you can specify source and destination, port, and protocol
- You may not create two security rules with the same priority and direction
- **Priority** could be a number between **100** and **4096**
- Rules are processed in priority order, with lower numbers processed before higher numbers, because lower numbers have higher priority

Understanding Network Security Groups

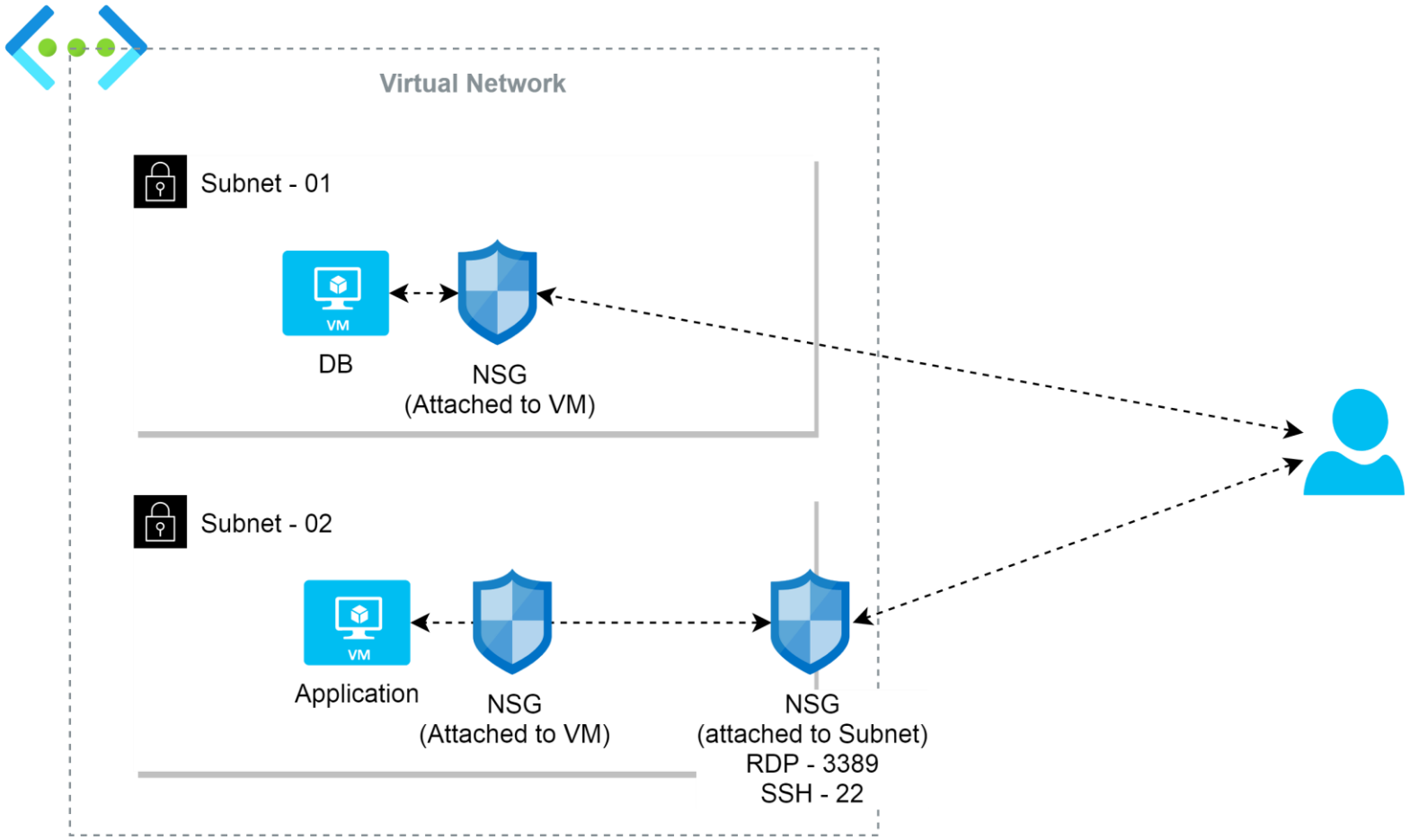
- Once traffic matches a rule, processing stops
- As a result, any rules that exist with lower priorities (higher numbers) that have the same attributes as rules with higher priorities are not processed

Ref: https://docs.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview

DenyAllInbound

Priority	Source	Source ports	Destination	Destination ports	Protocol	Access
65500	0.0.0.0/0	0-65535	0.0.0.0/0	0-65535	Any	Deny

Networks Security Groups (NSG): Use-cases



Network Security Group controls the Inbound and Outbound traffic.

Filter network traffic with a Network Security Group (NSG)

- You can use a network security group to filter network traffic inbound and outbound from a virtual network subnet
- Network security groups contain security rules that filter network traffic by IP address, port, and protocol
- Security rules are applied to resources deployed in a subnet

Networks Security Groups (NSG): Priority settings

Inbound port rules | Outbound port rules | Application security groups | Load balancing

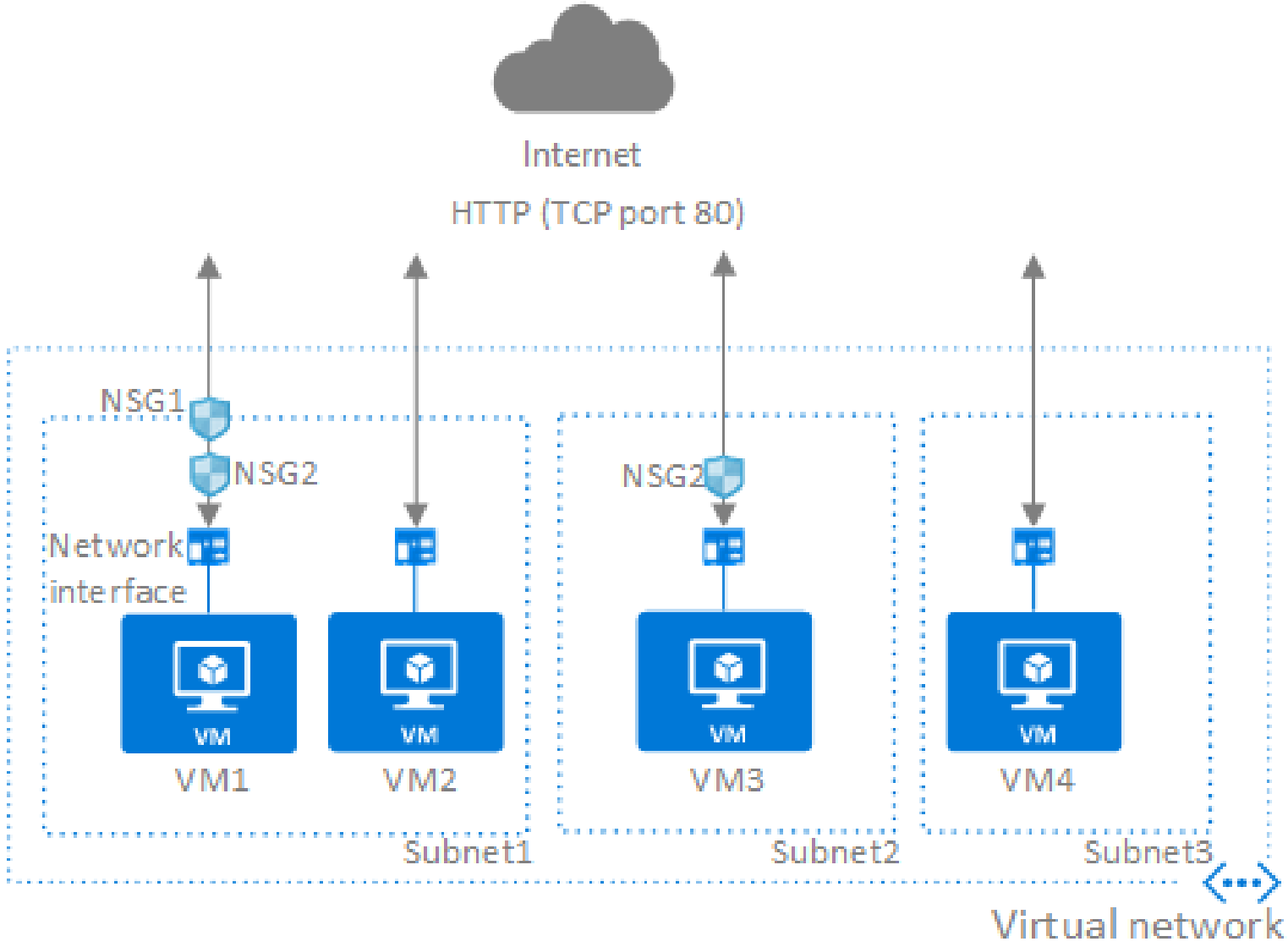
Network security group demovm-nsg (attached to network interface: demovm367)
Impacts 0 subnets, 1 network interfaces

Add inbound port rule

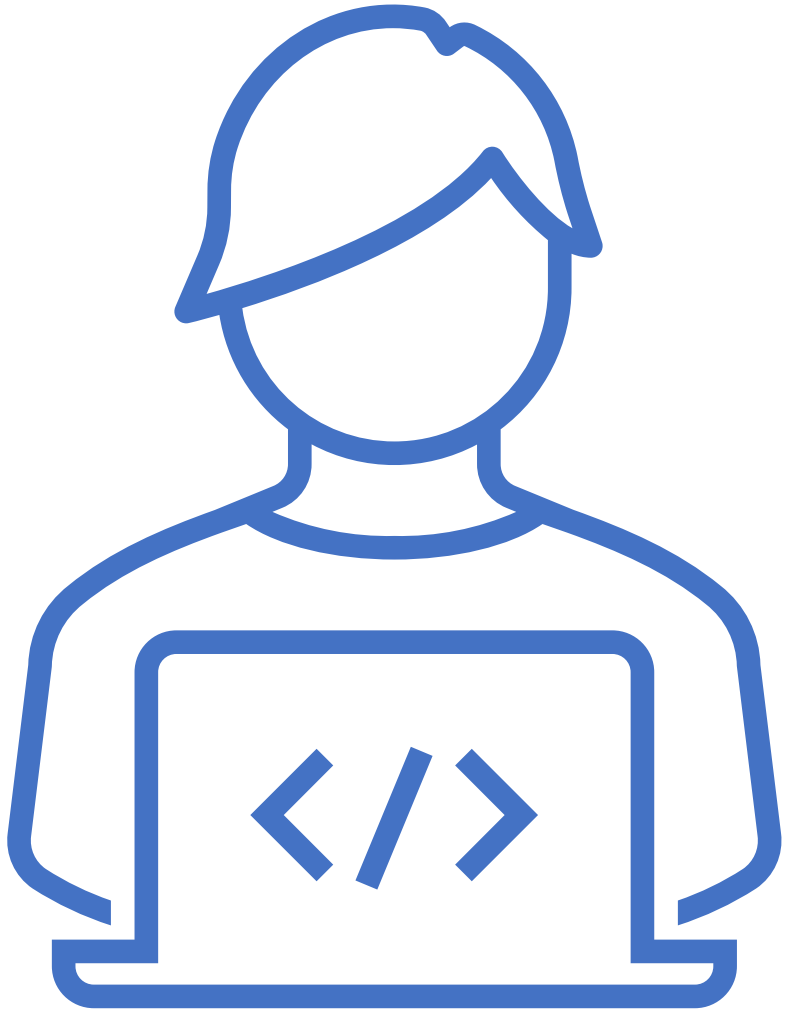
Priority	Name	Port	Protocol	Source	Destination
300	RDP	3389	TCP	92.98.34.198	Any
310	Port_80	80	TCP	Internet	Any
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwo
65001	AllowAzureLoadBalancerInB...	Any	Any	AzureLoadBalancer	Any
65500	DenyAllInBound	Any	Any	Any	Any

Ref: https://docs.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview

Filter Network traffic with a Network Security Group (NSG): Scenarios

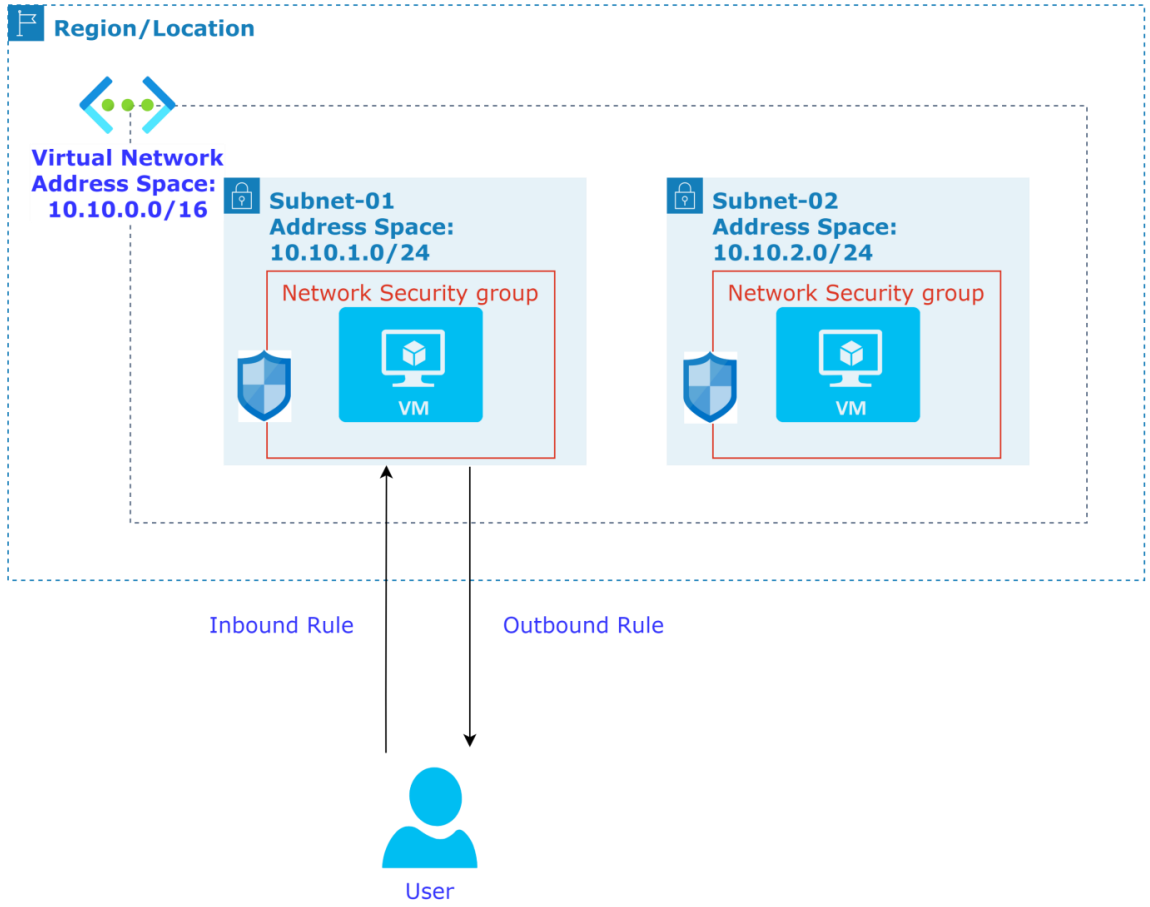


Hands-on Labs



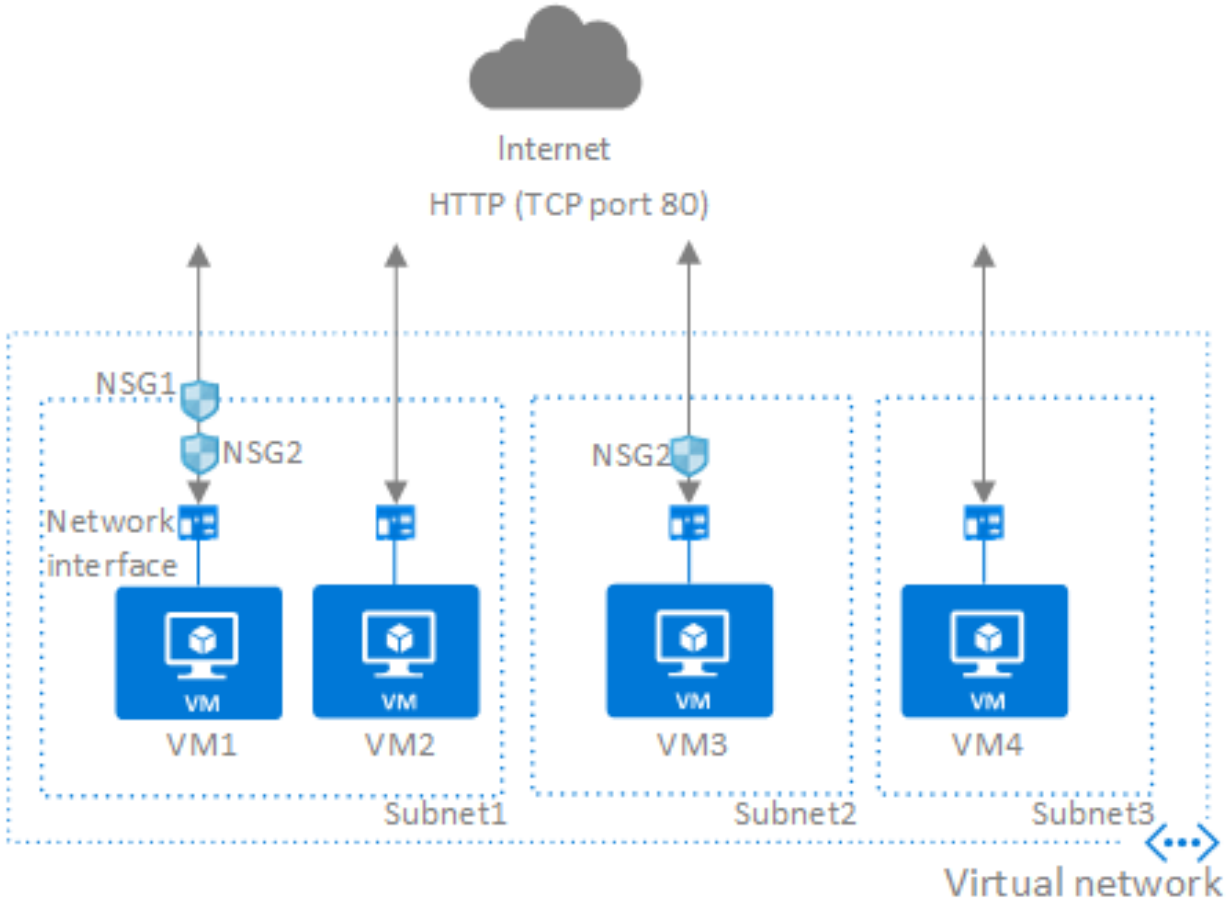
Lab: Create a Network Security Group and assign it to Azure VM

1. Create an Inbound rule to allow HTTP and RDP traffic for your public IP only
2. Then provision an Azure VM and assign the NSG create created in Step #1 to it



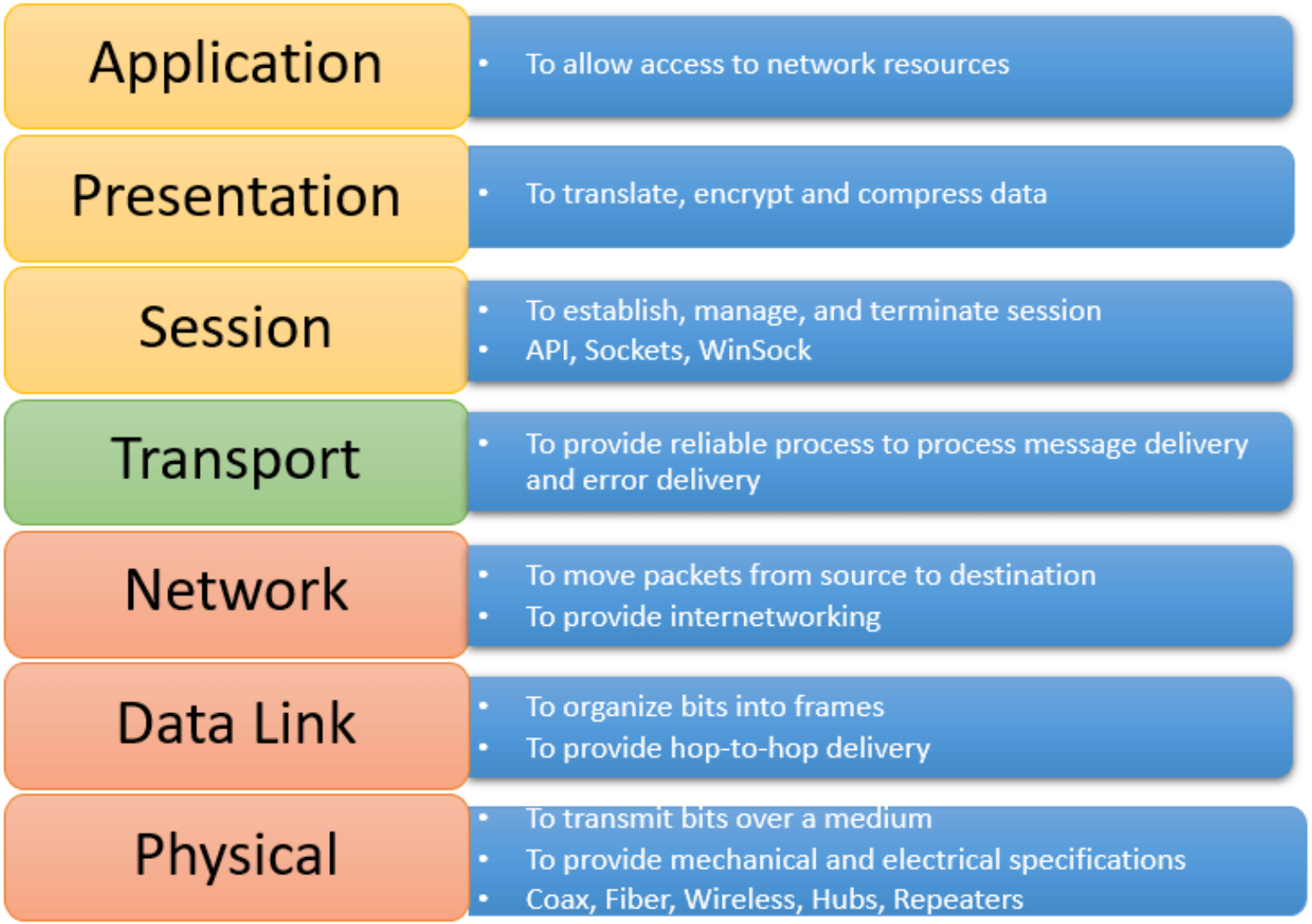
Lab: Filter Network traffic with a Network Security Group (NSG)

1. Create an Inbound rule to allow HTTP and RDP traffic for your public IP only
2. Then provision an Azure VM and assign the NSG create created in Step #1 to it



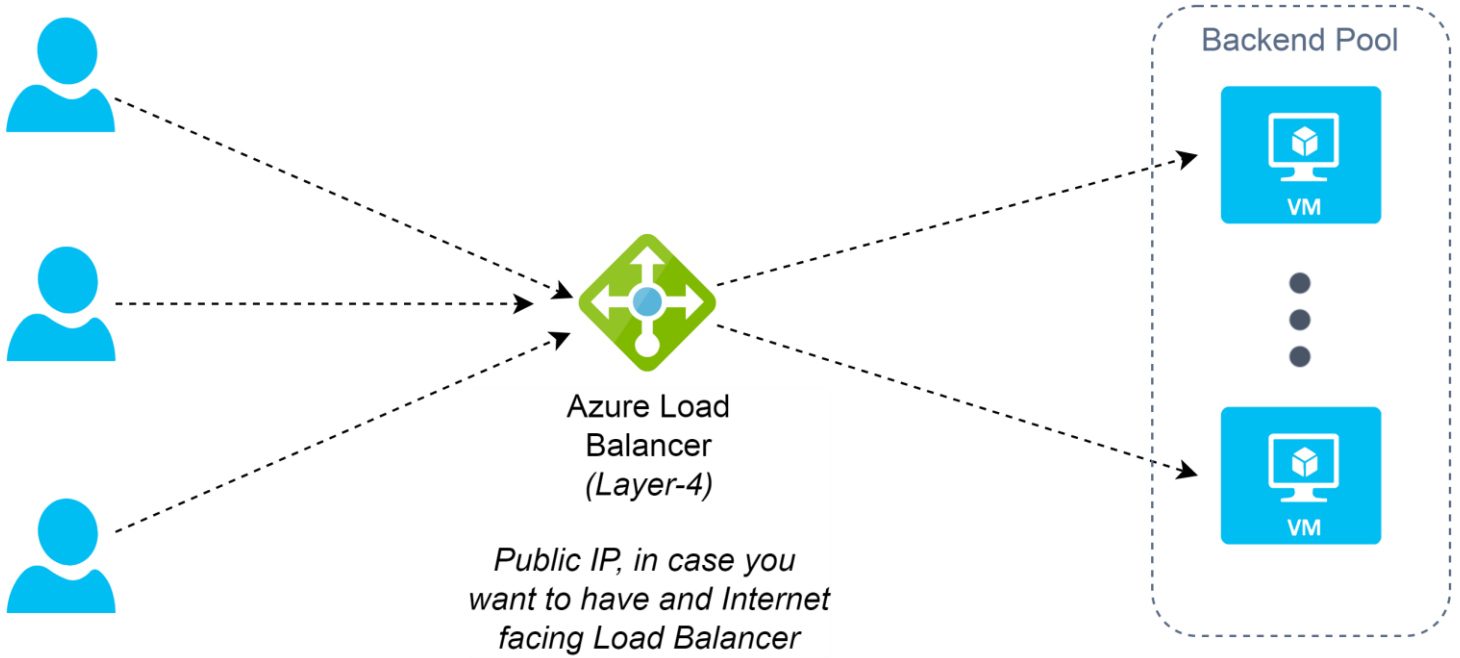
OSI Layers

Overview of OSI Layers

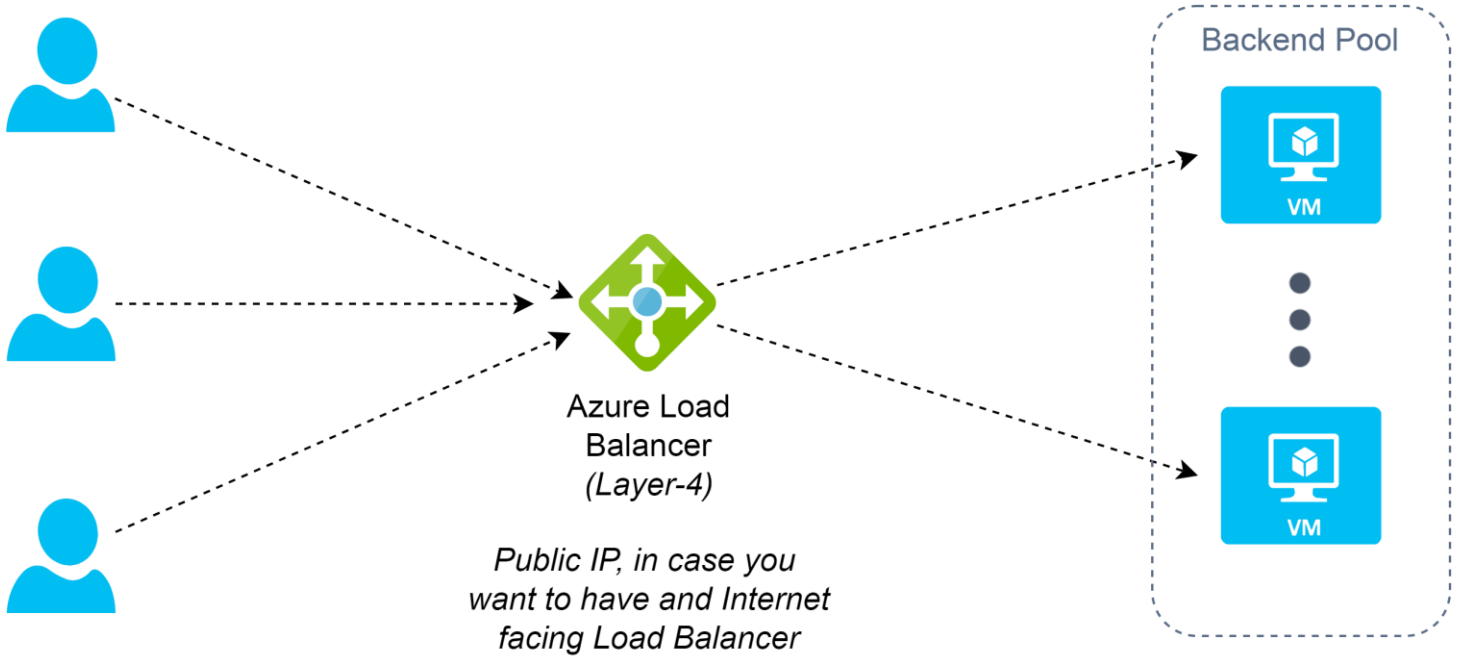


Azure Load Balancer service

Overview of Azure Load Balancer



Overview of Azure Load Balancer



Why use Azure Load Balancer?

- Load balance internal and external traffic to Azure virtual machines
- Increase availability by distributing resources within and across zones
- Configure outbound connectivity for Azure virtual machines
- Use health probes to monitor load-balanced resources
- Employ port forwarding to access virtual machines in a virtual network by public IP address and port
- Standard load balancer provides multi-dimensional metrics through Azure Monitor
- Standard load balancers and standard public IP addresses are closed to inbound connections unless opened by Network Security Groups
- NSGs are used to explicitly permit allowed traffic

Azure Load Balancer: Stock Keeping Units (SKU)

Azure Load Balancer has 3 SKUs - **Basic**, **Standard**, and **Gateway**. Each SKU is catered towards a specific scenario and has differences in **scale**, **features**, and **pricing**.

1. Basic Load Balancer
    - Equipped for small-scale applications that don't need high availability or redundancy
2. Standard Load Balancer
    - Equipped for load-balancing network layer traffic when high performance and ultra-low latency is needed
    - Routes traffic **within** and **across regions**, and to **availability zones** for **high resiliency**
3. Gateway Load Balancer (Preview)

Ref: https://docs.microsoft.com/en-us/azure/load-balancer/skus

Azure Load Balancer SKUs

	Standard Load Balancer	Basic Load Balancer
Scenario	Equipped for load-balancing network layer traffic when high performance and ultra-low latency is needed. Routes traffic within and across regions, and to availability zones for high resiliency.	Equipped for small-scale applications that don't need high availability or redundancy. Not compatible with availability zones.
Backend type	IP based, NIC based	NIC based
Protocol	TCP, UDP	TCP, UDP
Frontend IP Configurations	Supports up to 600 configurations	Supports up to 200 configurations
Backend pool size	Supports up to 1000 instances	Supports up to 300 instances
Backend pool endpoints	Any virtual machines or virtual machine scale sets in a single virtual network	Virtual machines in a single availability set or virtual machine scale set
Health probes	TCP, HTTP, HTTPS	TCP, HTTP
Health probe down behavior	TCP connections stay alive on an instance probe down and on all probes down.	TCP connections stay alive on an instance probe down. All TCP connections end when all probes are down.
Availability Zones	Zone-redundant and zonal frontends for inbound and outbound traffic	Not available

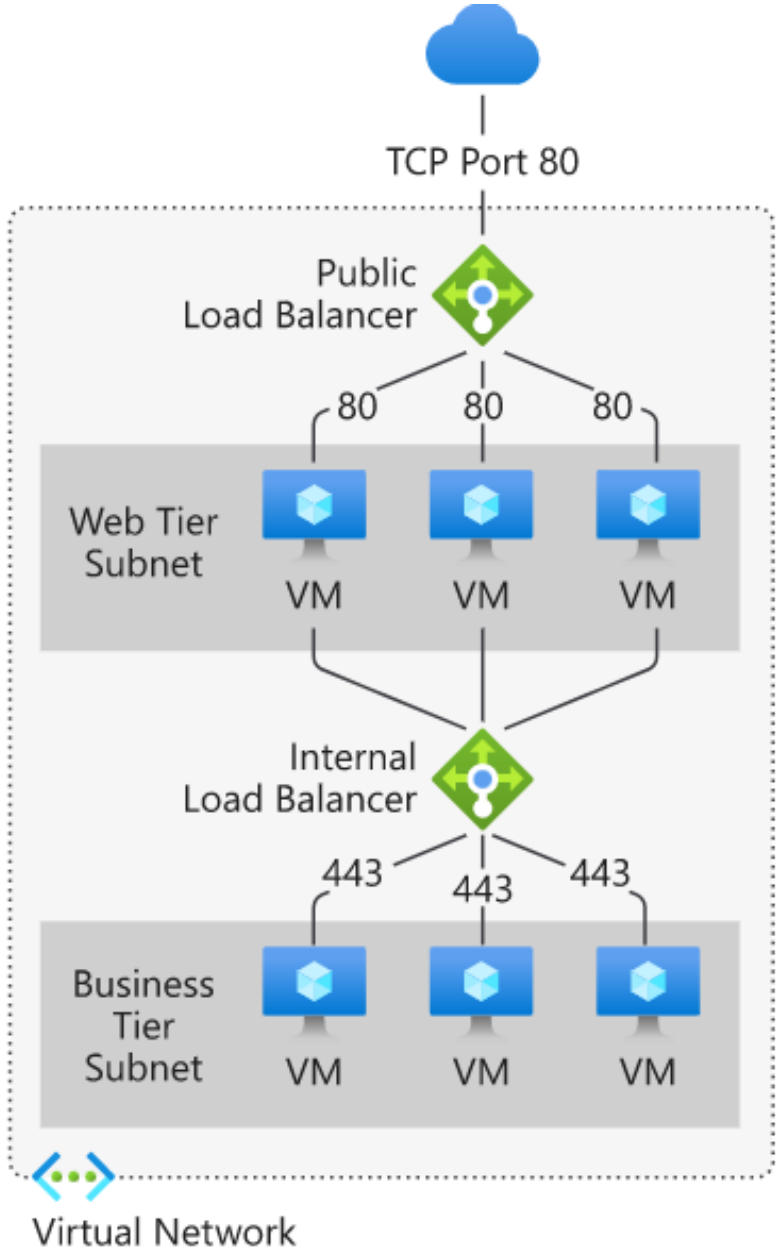
Azure Load Balancer: Important Points

- ✓ Microsoft recommends **Standard load balancer**
- ✓ Standalone **VMs**, **availability sets**, and **virtual machine scale sets** can be connected to only one Load Balancer SKU, never both
- ✓ **Load balancer** and the **public IP address SKU** must match when you use them with public IP addresses Load balancer and public IP SKUs aren't mutable

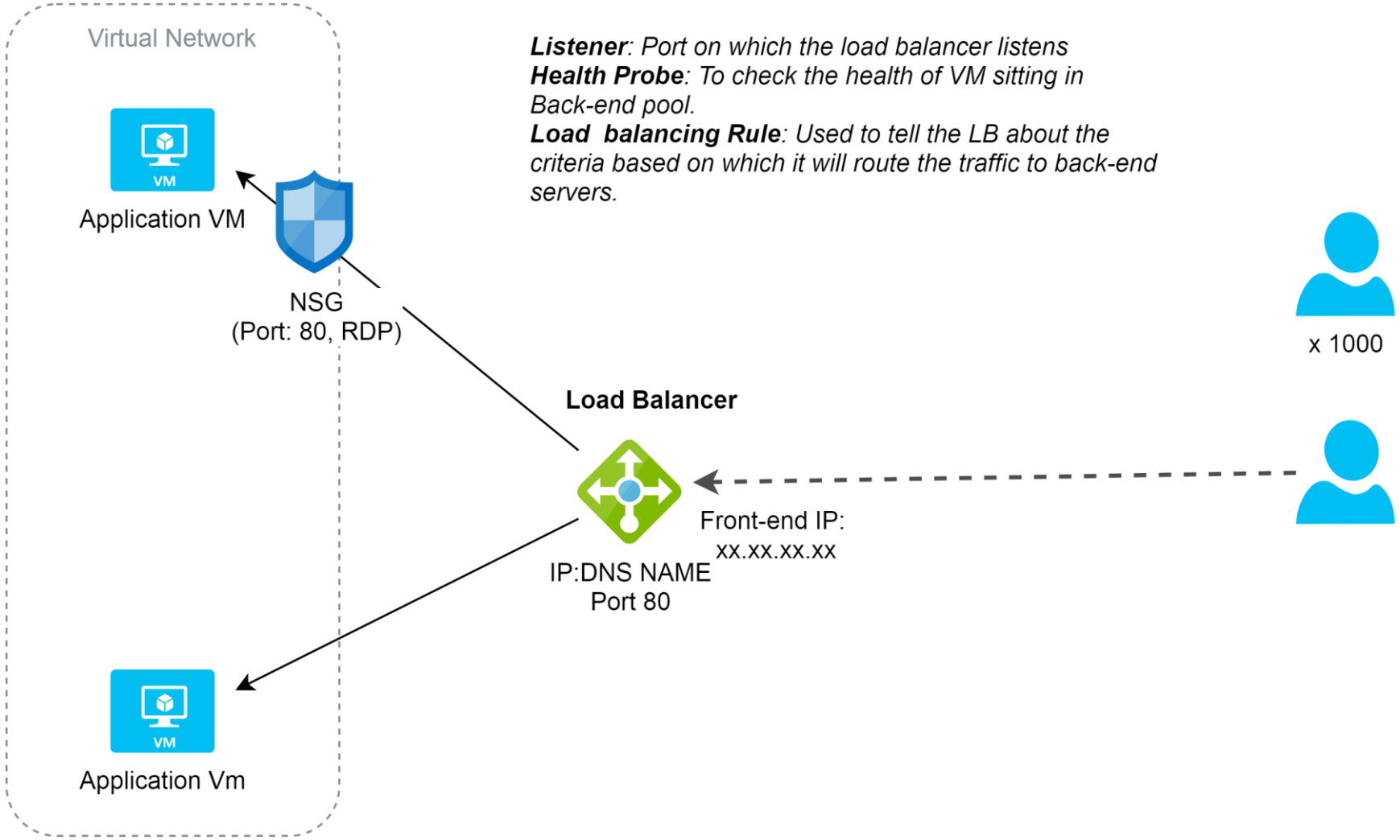
Types of Azure Load Balancer (Layer-4)

1. Public Load Balancer (External)
   - A public load balancer can provide outbound connections for virtual machines (VMs) inside your virtual network
   - These connections are accomplished by translating their private IP addresses to public IP addresses
   - Public Load Balancers are used to load balance internet traffic to your VMs
2. Internal Load balancer
   - An internal (or private) load balancer is used where private IPs are needed at the frontend only
   - Internal load balancers are used to load balance traffic inside a virtual network
   - A load balancer frontend can be accessed from an on-premises network in a hybrid scenario

Internal and Public Load Balancer (Layer-04)



Azure Load Balancer: Main Components



***Listener**: Port on which the load balancer listens*
***Health Probe**: To check the health of VM sitting in Back-end pool.*
***Load balancing Rule**: Used to tell the LB about the criteria based on which it will route the traffic to back-end servers.*

Azure Load Balancer: Health Probe

- When using load-balancing rules with Azure Load Balancer, you need to specify health probes to allow Load Balancer to detect the backend endpoint status.
- The configuration of the health probe and probe responses determine which backend pool instances will receive new flows.
- You can use health probes to detect the failure of an application on a backend endpoint.

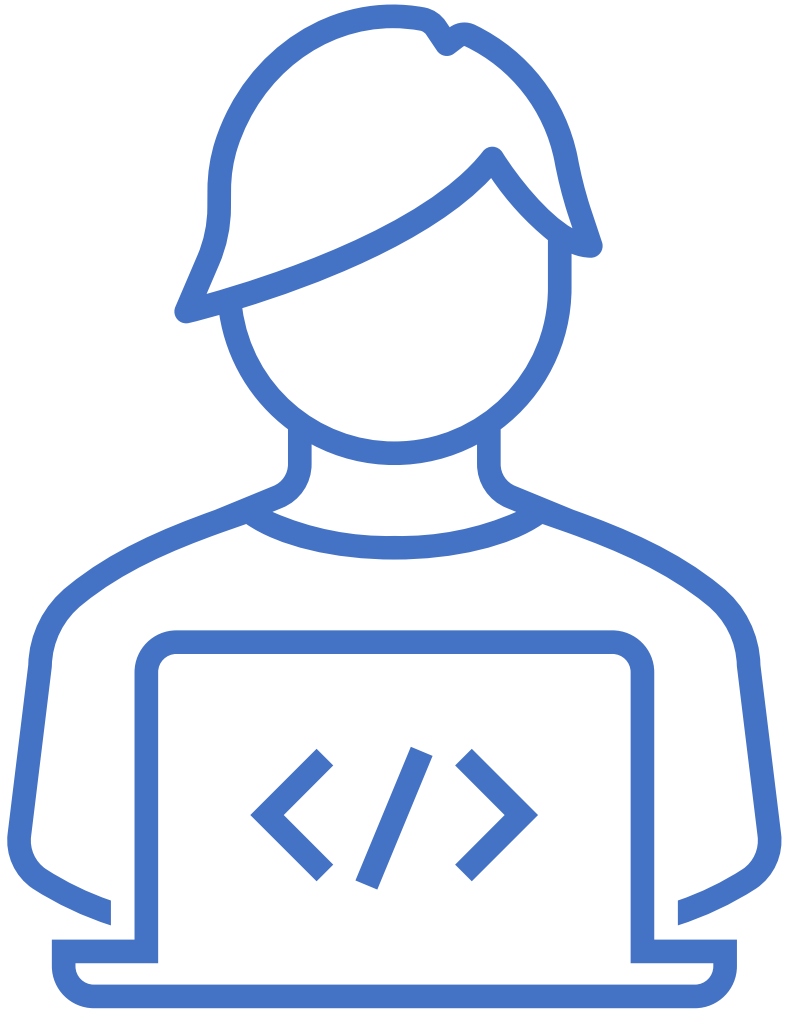
Ref: https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-custom-probe-overview

	Standard SKU	Basic SKU
Probe types	TCP, HTTP, HTTPS	TCP, HTTP
Probe down behavior	All probes down, all TCP flows continue.	All probes down, all TCP flows expire.

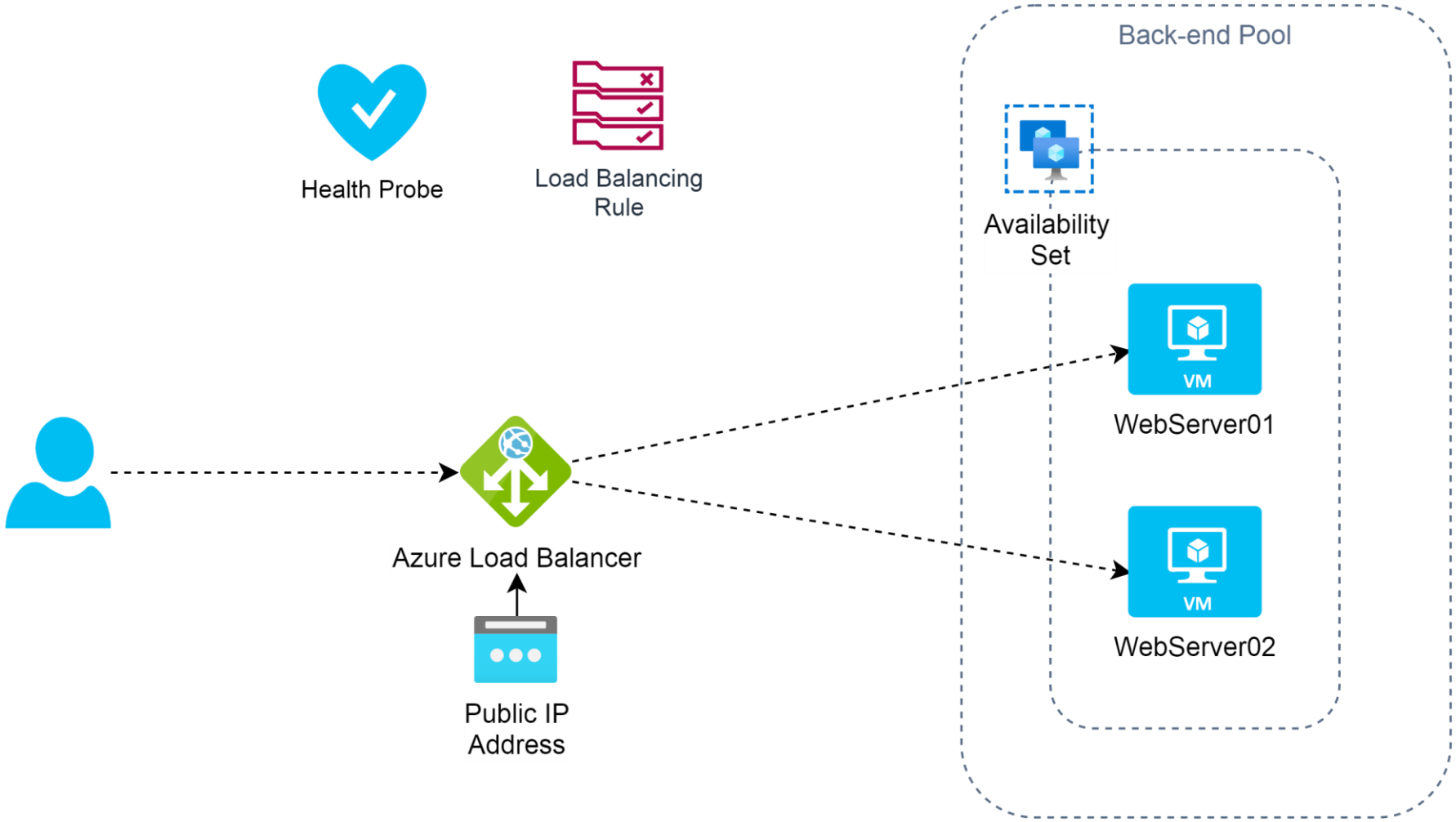
Azure Load Balancer: NAT Rule

- NAT rules are used to specify a backend resource to route traffic to
- For example, configuring a specific load balancer port to send RDP traffic to a specific VM
- Load-balancing rules are used to specify a pool of backend resources to route traffic to, balancing the load across each instance
- For example, a load balancer rule can route TCP packets on port 80 of the load balancer across a pool of web servers

Hands-on Labs



Lab: Create a Load Balancer with Basic SKU



Lab: Create a Load Balancer with Basic SKU

Pre-requisites:

- *Create a new Availability set (will provision the VMs in this)*
- *Create a new Security Group with ports 80 and 3389 open*
- *Create a new VNet with two subnet (will provision the VM in this)*

1. Provision two VMs and install IIS role on both the VMs
2. Put a simple HTML page as the home page on the VMs
3. Create a Public IP Address
4. Create an Azure Load Balancer
5. Create a Backend pool and add both the VM in it
6. Add a Health Probe
7. Add Load Balancing rules

Lab: Create a Load Balancer with Standard SKU

Scenario: Create an Azure Load balancer with below configuration:

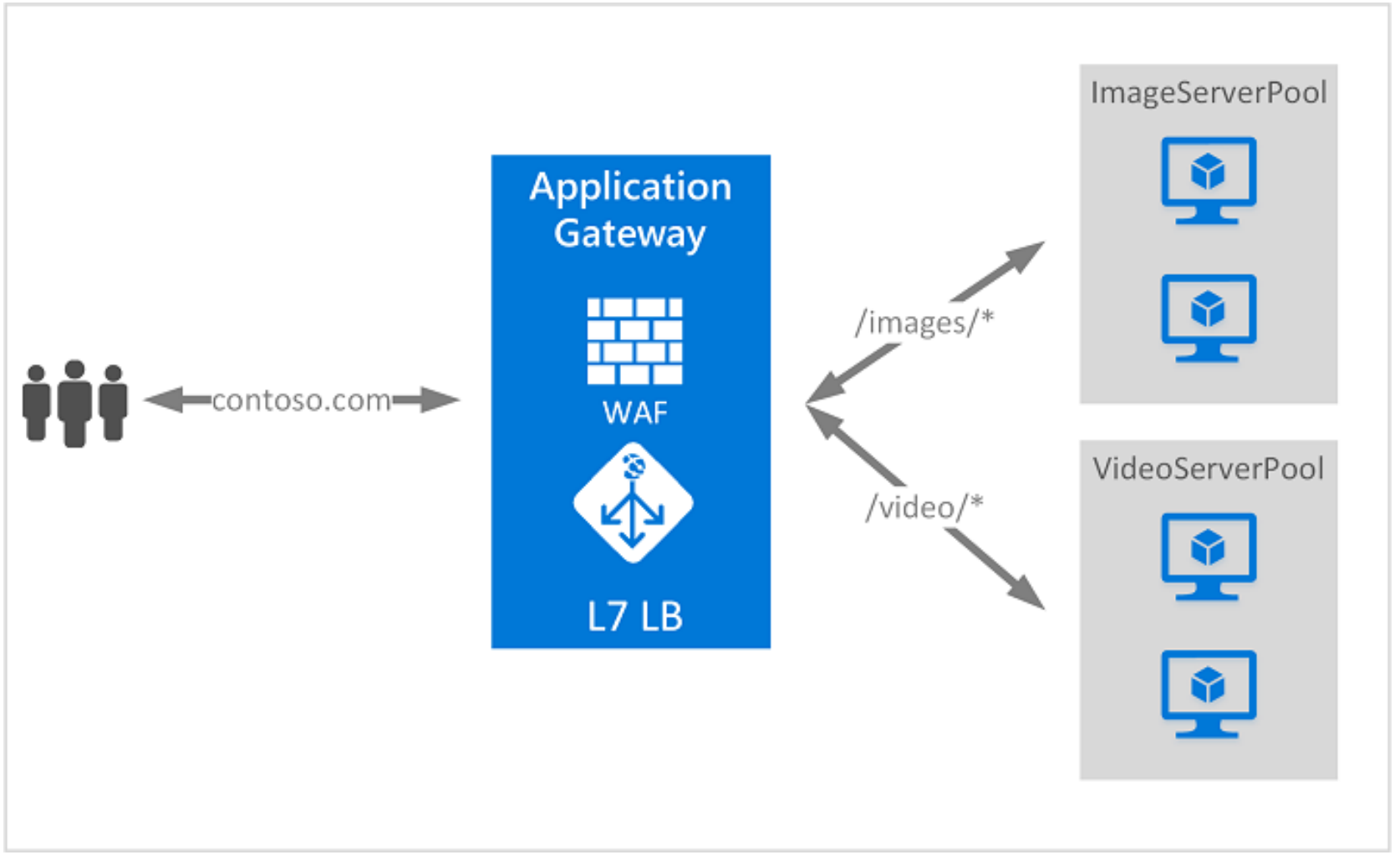
1) Create one back-end pool with two Web Servers (VMs with any webserver role installed)
2) Create a simple html page with the server IP address detail in it and make it a landing page of that web server
3) Add a health probe
4) Create a front-end IP address for load balancer
5) Load balancer rule: Listen on port 80

Azure Application Gateway

Overview of Azure Application Gateway service

- Azure Application Gateway is a web traffic load balancer that is distribute traffic to web-applications
- The applications can reside inside Virtual Machines, VM Scale-sets or On-premise servers
- The Application Gateway is an OSI Layer-7 load balancing service
- Secure Socket Layer (SSL/TLS) termination
- The request to the backend pool from Application gateway can go unencrypted
- This can lift the burden of the backend pool for decrypting the requests

Azure Application Gateway: Conceptual Diagram



Azure Application Gateway: Key Points

- Application Gateway is **web traffic load balancer**
- It allows you to distribute incoming traffic based on HTTP request properties such as URL and host headers
- Application gateway has four tiers: **Standard, Standard V2, WAF, and WAF v2**
- You can use the same application gateway for up to 100+ websites with multi-site hosting
- Set the minimum and maximum scale units based on your needs
- Azure Application Gateway vs Azure Load Balancer
  - An **application gateway** operates at layer 7
  - A **load balancer** functions at layer 4
- You can use both public and private IP on the frontend

Azure Application Gateway: Key Points

- You can also enable ***Autoscaling*** for your Application Gateway resource
- This allows the Application Gateway to ***scale-out*** or ***scale-in*** based on the traffic load pattern
- You can also enable the ***Web Application Firewall (WAF)*** feature for the Application Gateway resource
- You can also enable ***session affinity*** which allows a user session to directed to the same server for processing. If the state of the user session is stored on the server, then this feature can be very useful

Components of Azure Application Gateway

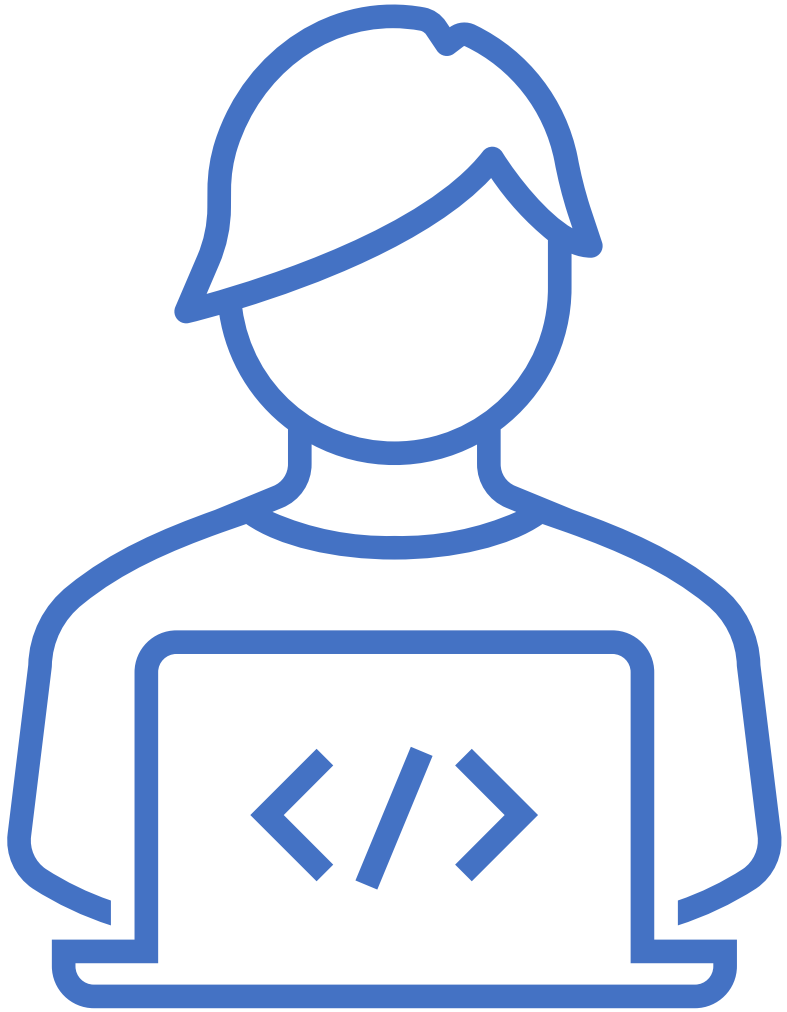
- **Front-end IP Address**: Users will hit the Application Gateway via the front-end IP address
- **Listener**: This is a logical entity that checks for incoming connection requests. There are can be multiple listeners attached to the Application Gateway
- There are basically two types of Listener configuration:
  1) **Basic**: Here the listener listens to a single domain site
  2) **Multi-site**: Here the listener maps to multiple domain site
- **Routing Rules**: This is used to route the traffic from the listener to the backend pool

  There are two types of routing rule:
  1) **Basic**: All the request are routed to the backend pool directly
  2) **Path-based**: Request are routed to the backend pool based on the URL in the request

Components of Azure Application Gateway

- **Backend Pool**: These can be Network Interface Cards, Virtual Machine Scale sets, Public or Internal IP addresses, FQDN or backend such as App service
- **Health Probe**: This defines how the application gateway will monitor the health of the resources in the backend pool

Hands-on Labs



Azure Traffic Manager

Overview of Azure Traffic Manager

- Azure Traffic Manager is a DNS-based traffic load balancer
- This service allows you to distribute traffic to your public facing applications across the global Azure regions
- Traffic Manager also provides your public endpoints with high availability and quick responsiveness
- Traffic Manager uses DNS to direct the client requests to the appropriate service endpoint based on a traffic-routing method
- Traffic manager also provides health monitoring for every endpoint
- The endpoint can be any Internet-facing service hosted inside or outside of Azure

Azure Traffic Manager: Traffic Routing Methods

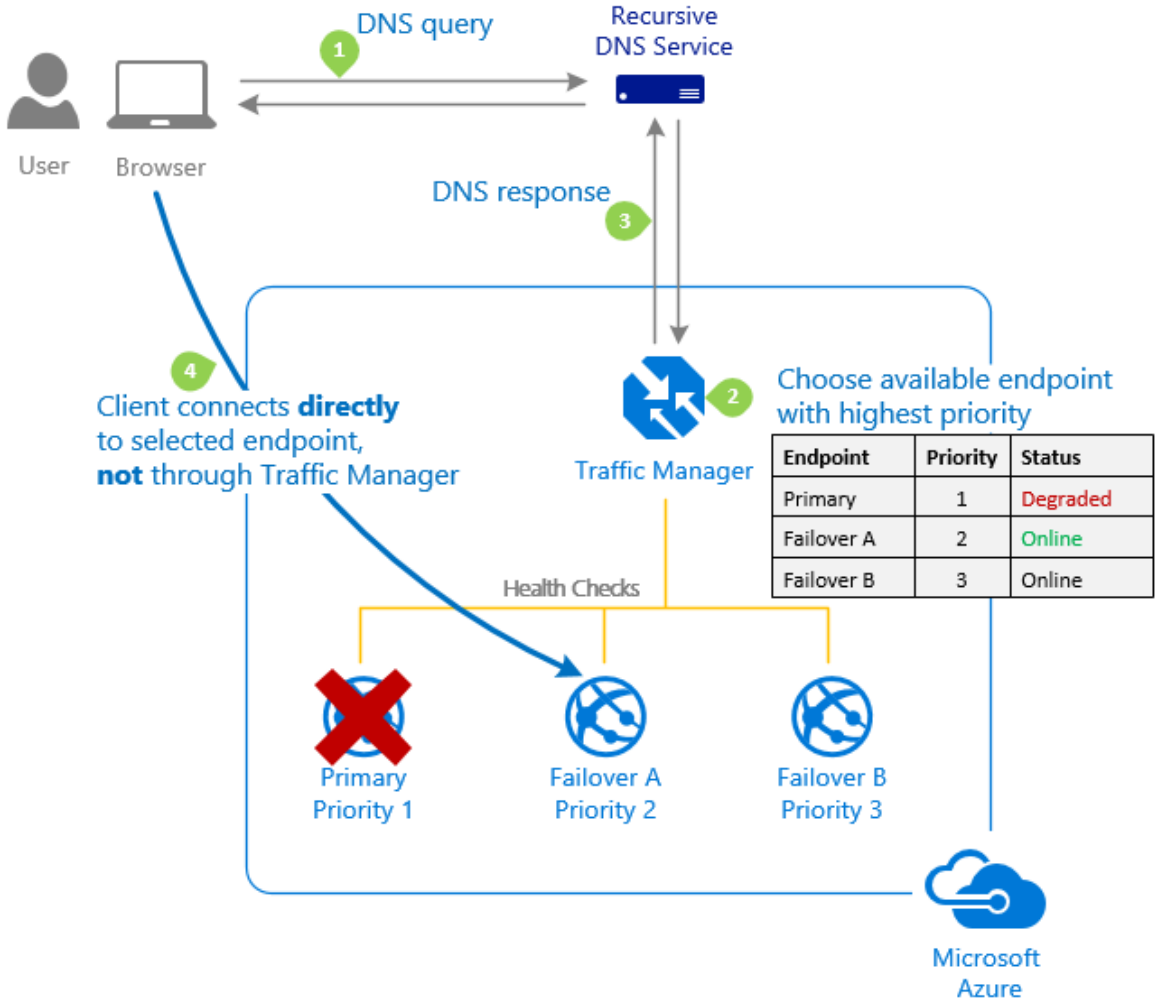
- Azure Traffic Manager supports six traffic-routing methods to determine how to route network traffic to the various service endpoints:
  1) **Priority**: Select Priority routing when you want to have a primary service endpoint for all traffic. You can provide multiple backup endpoints in case the primary or one of the backup endpoints is unavailable
  2) **Weighted**: Select Weighted routing when you want to distribute traffic across a set of endpoints based on their weight. Set the weight the same to distribute evenly across all endpoints
  3) **Performance**: Select Performance routing when you have endpoints in different geographic locations, and you want end users to use the "closest" endpoint for the lowest network latency

Azure Traffic Manager: Traffic Routing Methods

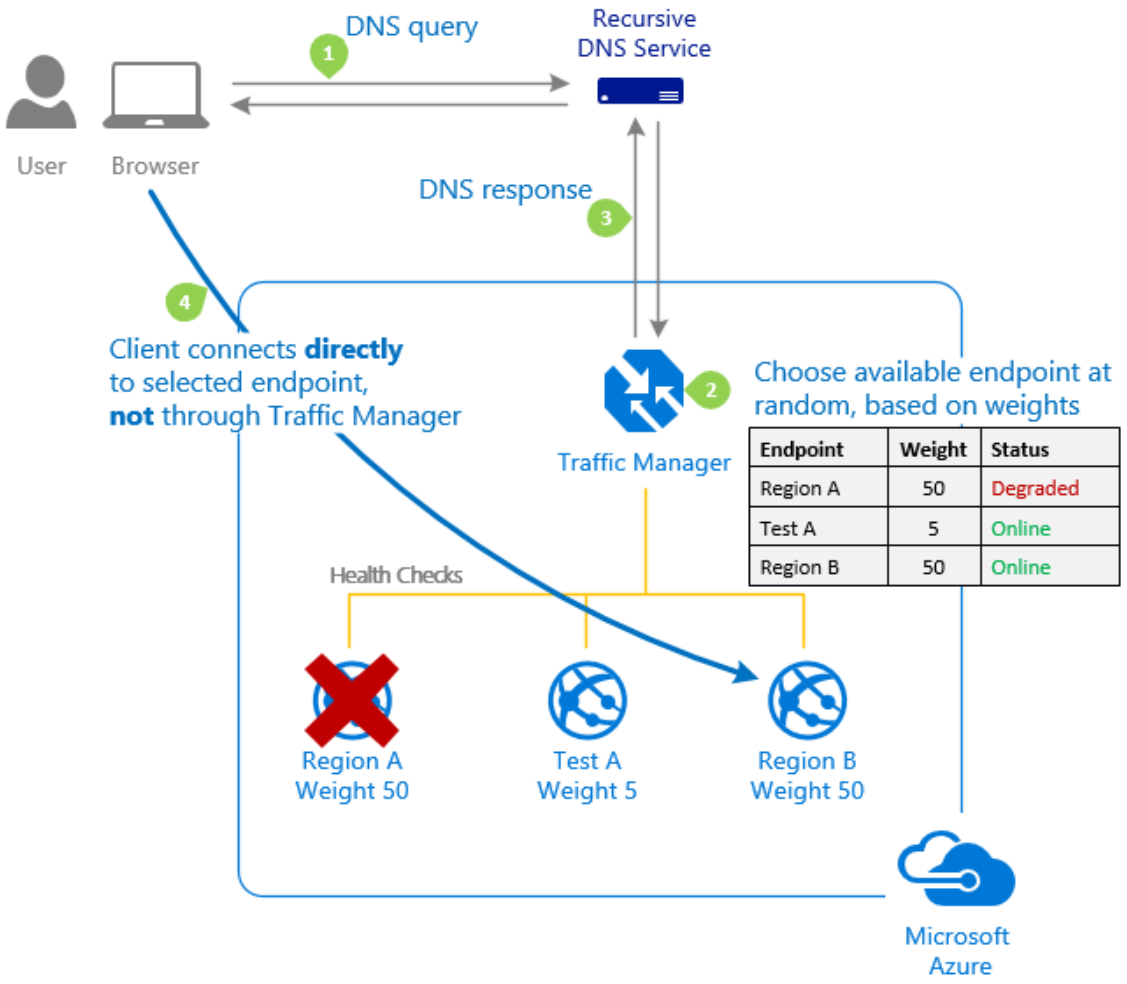
4) **Geographic**: Select Geographic routing to direct users to specific endpoints (Azure, External, or Nested) based on where their DNS queries originate from geographically. With this routing method, it enables you to be in compliance with scenarios such as data sovereignty mandates, localization of content & user experience and measuring traffic from different regions
5) **Multivalue**: Select MultiValue for Traffic Manager profiles that can only have IPv4/IPv6 addresses as endpoints. When a query is received for this profile, all healthy endpoints are returned
6) **Subnet**: Select Subnet traffic-routing method to map sets of end-user IP address ranges to a specific endpoint. When a request is received, the endpoint returned will be the one mapped for that request's source IP address

Ref: https://docs.microsoft.com/bs-latn-ba/azure/traffic-manager/traffic-manager-routing-methods

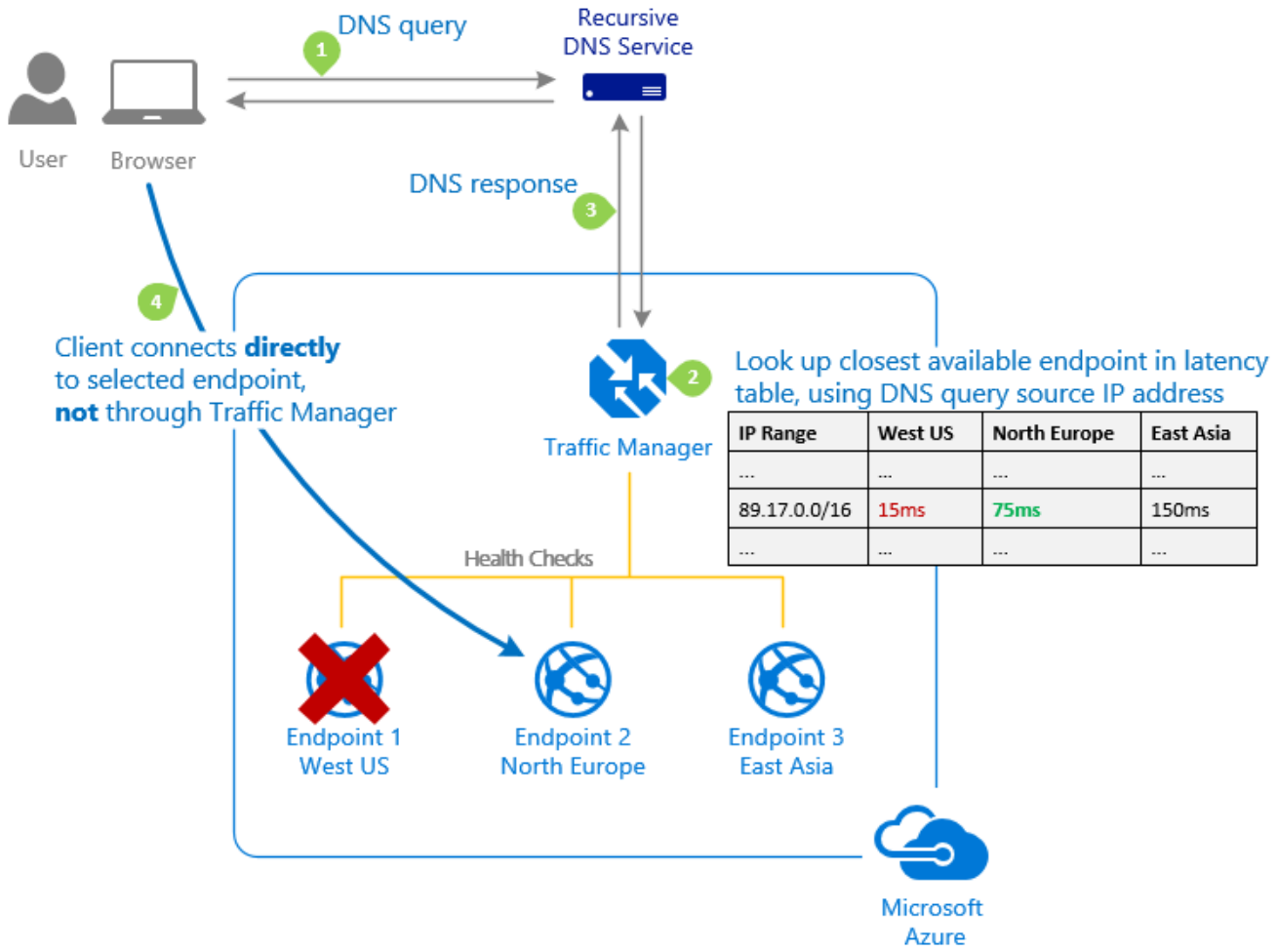
Priority Traffic Routing Method



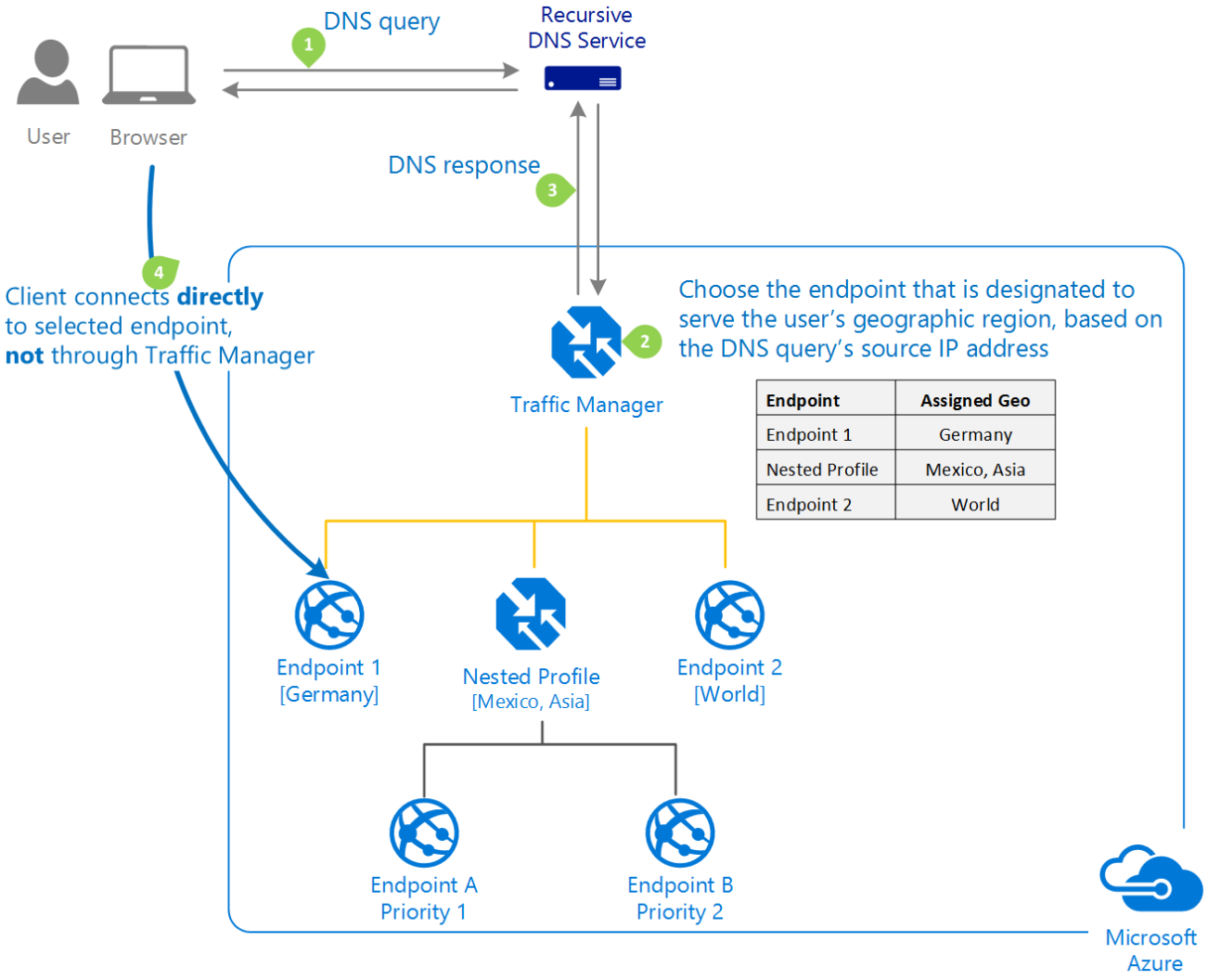
Weighted Traffic Routing Method



Performance Traffic Routing Method



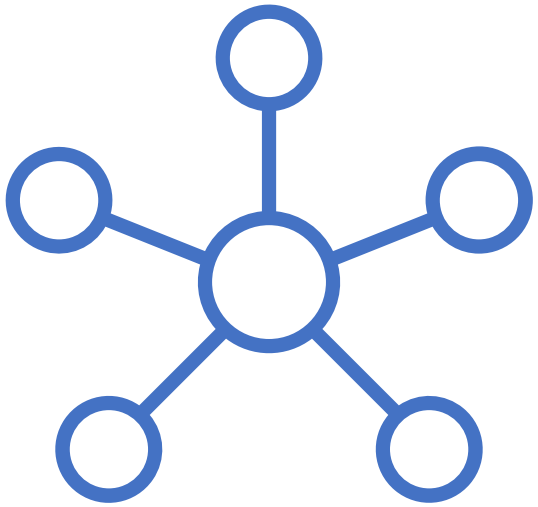
Geographic Traffic Routing Method



Endpoint	Assigned Geo
Endpoint 1	Germany
Nested Profile	Mexico, Asia
Endpoint 2	World

Azure Load Balancer vs Application Gateway vs Traffic Manager

	Load Balancer	Application Gateway	Traffic Manager
Service	Network load balancer.	Web traffic load balancer.	DNS-based traffic load balancer.
Network Protocols	Layer 4 (TCP or UDP)	Layer 7 (HTTP/HTTPS)	Layer 7 (DNS)
Type	Internal and Public	Standard and WAF	–
Routing	Hash-based, Source IP affinity	Path-based	Performance, Weighted, Priority, Geographic, MultiValue, Subnet
Global/Regional Service	Global/Regional	Regional	Global
Recommended Traffic	Non-HTTP(S)	HTTP(S)	Non-HTTP(S)
Endpoint Monitoring	Health probes	Health probes	HTTP/HTTPS GET requests
Redundancy	Zone redundant and Zonal	Zone redundant	Resilient to regional failures
SSL/TLS Termination	–	Supported	–
Sticky Sessions	Supported	Supported	–
VNet Peering	Supported	Supported	–
SKU	Basic and Standard	Standard and WAF (v1 & v2)	–
Pricing	Standard Load Balancer – charged based on the number of rules and processed data.	Charged based on Application Gateway type, processed data, outbound data transfers, and SKU.	Charged per DNS queries, health checks, measurements, and processed data points.



Virtual Network Components

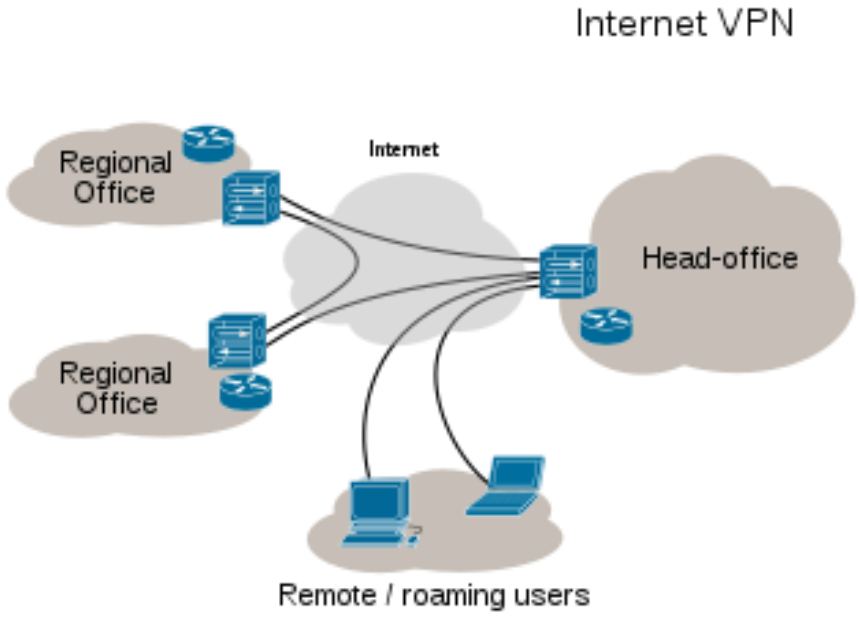
Virtual Network Components

1. **NAT Gateway**
   - Allows your virtual network resources to have an outbound-only connection
   - A NAT gateway resource can use up to 16 static IP addresses
   - You can use multiple subnets in a NAT gateway
2. **Route tables** are used to determine where network traffic is directed
   - A subnet can only be associated with one route table
   - If multiple routes contain the same address prefix, the selection will be based on the following priority: User-defined route, BGP route, and System route
3. You can connect VNets to each other using **VNet peering**
4. If you need to connect privately to a service, you can use **Azure Private Endpoint** powered by Azure Private Link

What is a Virtual Private Network? (VPN)

What is Virtual Private Network (VPN)?

1. A virtual private network, or VPN, is an encrypted connection over the Internet from a device to a network
2. The encrypted connection helps ensure that sensitive data is safely transmitted
3. It prevents unauthorized people from eavesdropping on the traffic and allows the user to conduct work remotely
4. VPN technology is widely used in corporate environments



Virtual Network: Connectivity Services

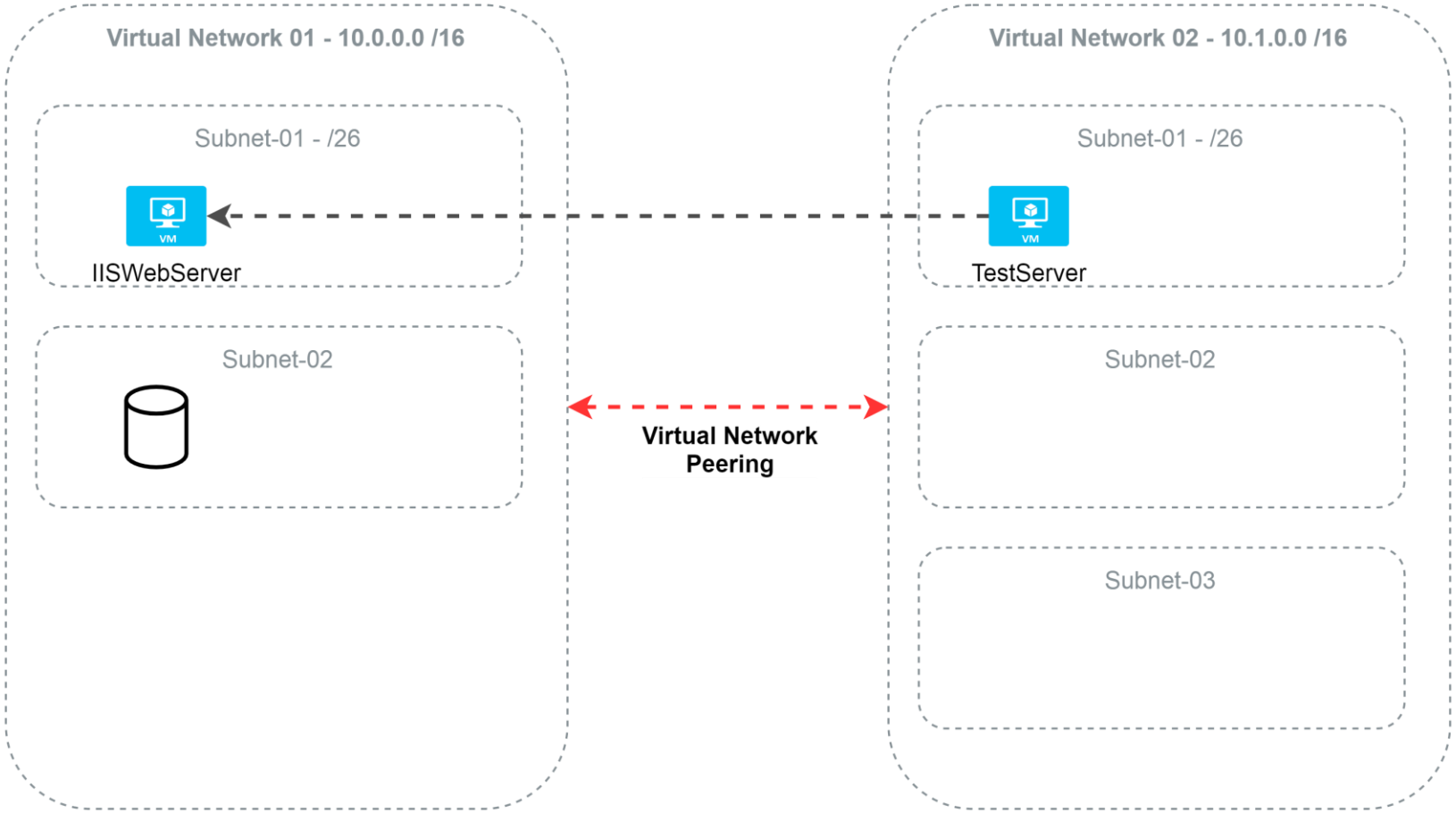
Azure Network: Connectivity options

1. **VNet to VNet** (VNet Peering)
   - Virtual Network Peering (both the Networks in the same)
   - Global VNet Peering (Networks across different Azure regions)
2. **Point to Site VPN** (P2S)
3. **Site to Site VPN** (S2S)
4. **ExpressRoute**

Overview of Virtual Network Peering

- Virtual network peering enables you to seamlessly connect two or more Virtual Networks in Azure
- The traffic between virtual machines in peered virtual networks uses the **Microsoft backbone infrastructure**
- You can:
  - Connect virtual networks in the same Azure region known as virtual network peering
  - Connect virtual networks across different Azure regions known as global virtual network peering
- Ensure that your VNet address ranges do not overlap with one another. Plan accordingly before initiating the peer

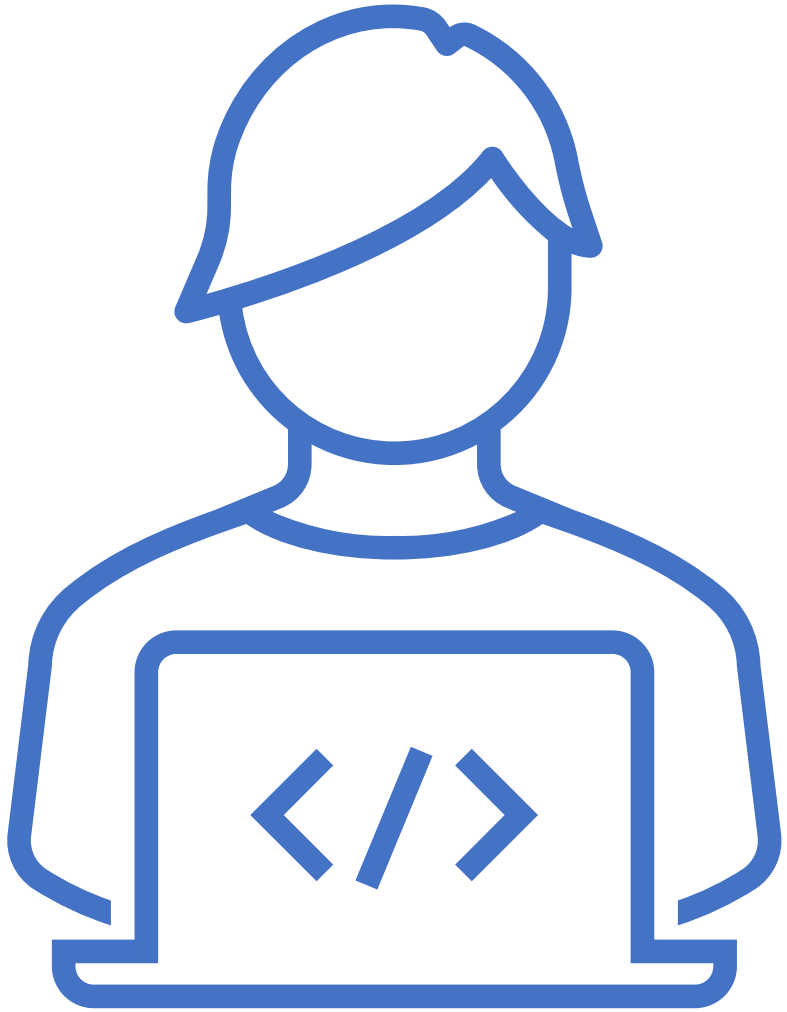
Overview of Virtual Network Peering



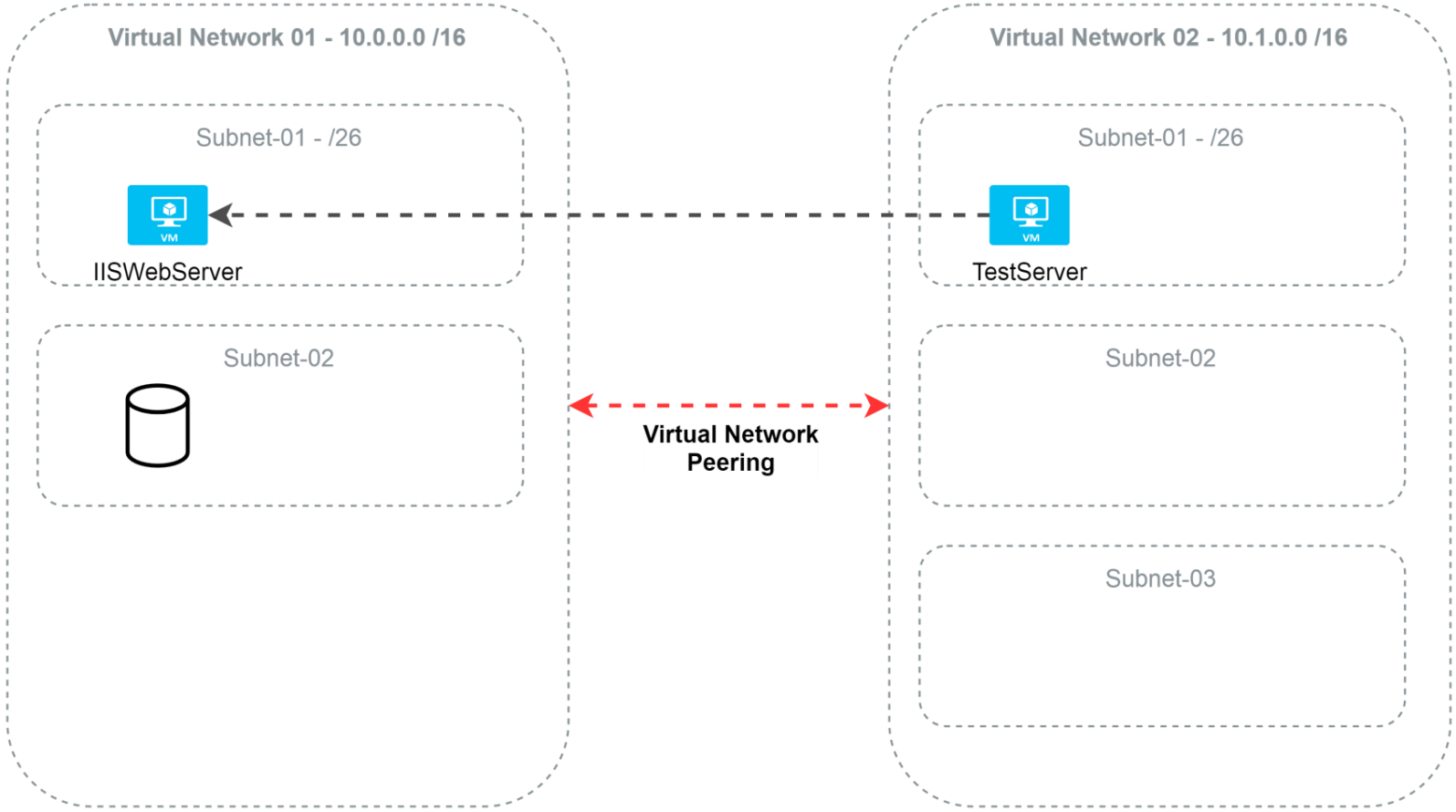
Virtual Network Peering: Important Notes

- 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 you enable virtual network peering between two virtual networks, the virtual machines can then communicate via their private IP addresses across the peering connection
- You can also peer virtual networks that are located across different subscriptions
- The virtual networks can't have overlapping CIDR blocks

Hands-on Labs



Lab: Connect virtual networks with virtual network peering

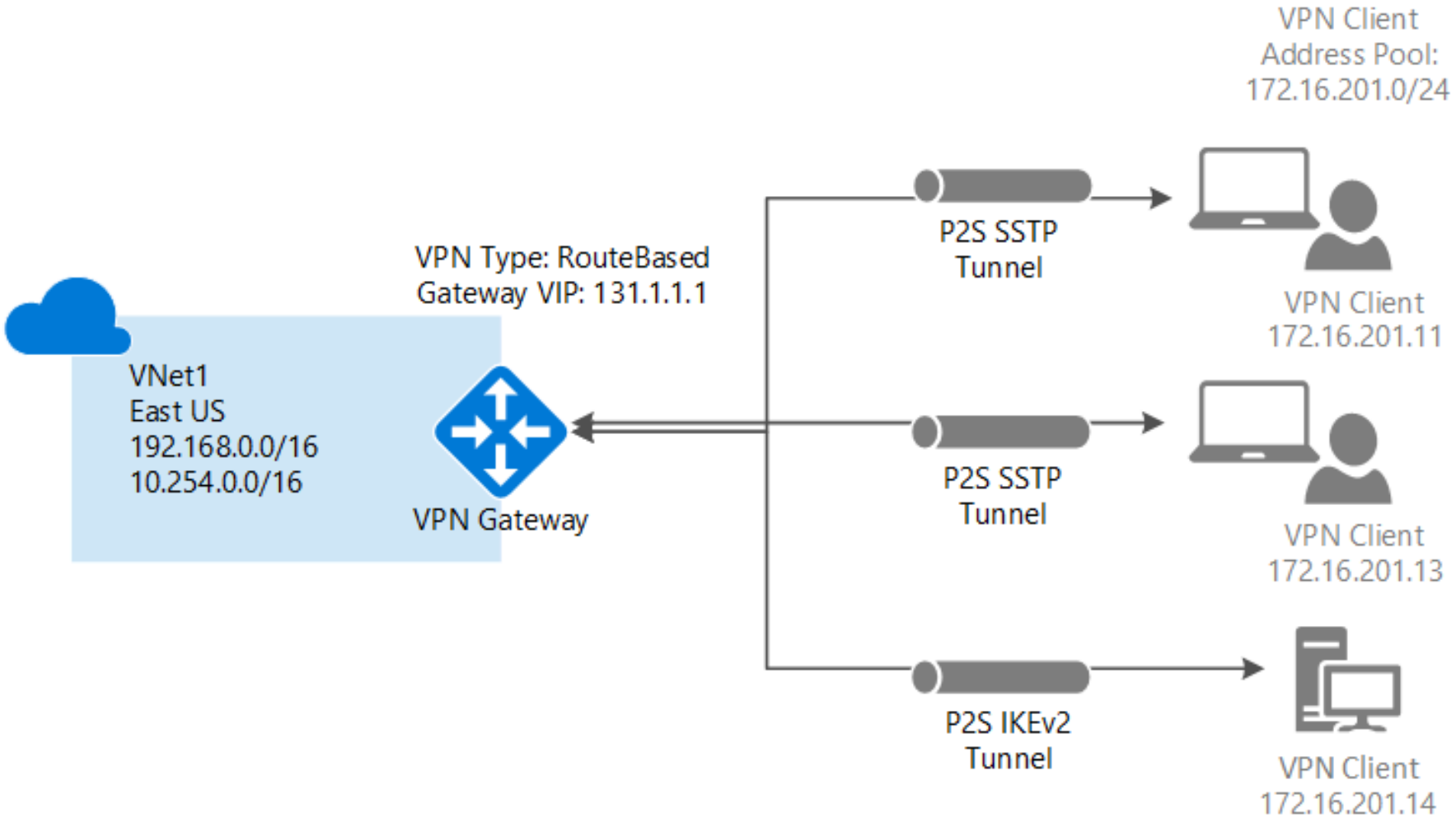


Point-to-Site VPN Connectivity

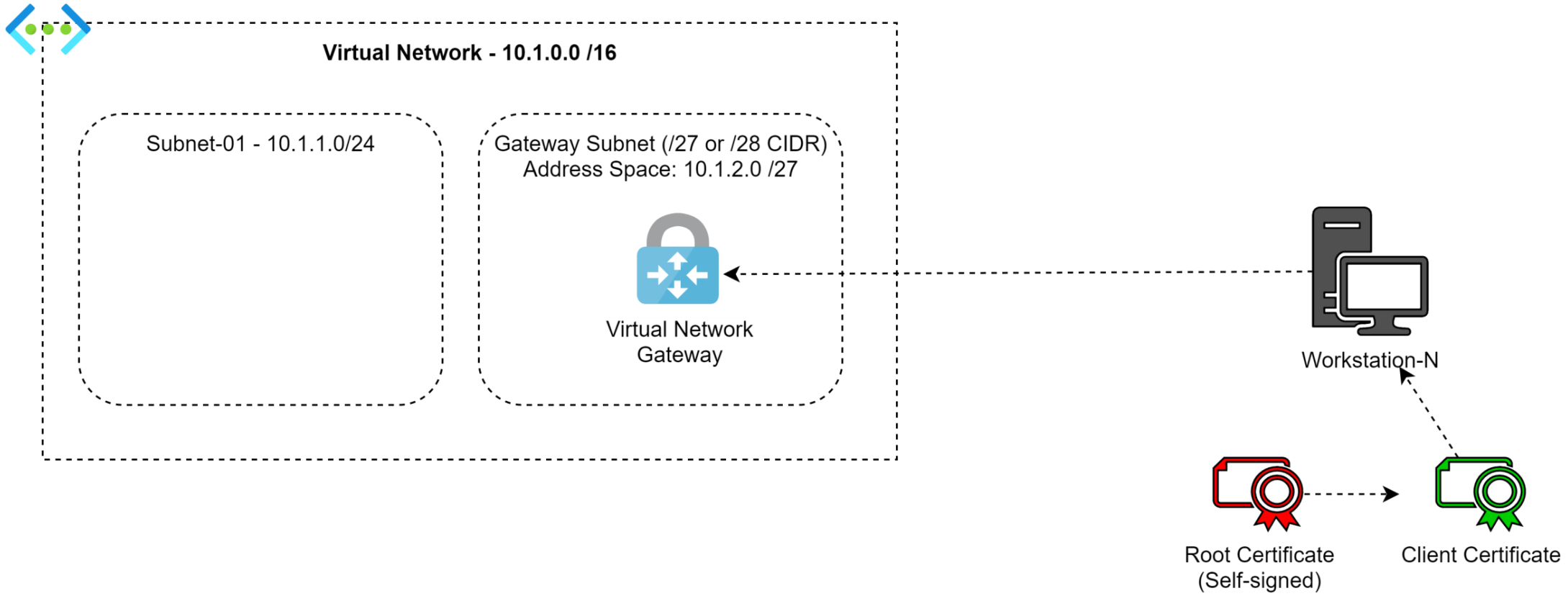
- A Point-to-Site (P2S) VPN gateway connection lets you create a secure connection to your virtual network from an **individual client pc**
- A P2S connection is established by starting it from the client computer
- This solution is useful for telecommuters who want to connect to Azure VNets from a remote location, such as from home or a conference
- P2S VPN is also a useful solution to use instead of S2S VPN when you have only a few clients that need to connect to a Vnet

Ref: https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-certificates-point-to-site

Point-to-Site VPN Connectivity



Point-to-Site VPN Connectivity



Azure VPN Gateway: Pricing and SKUs

Refer below link for the details:

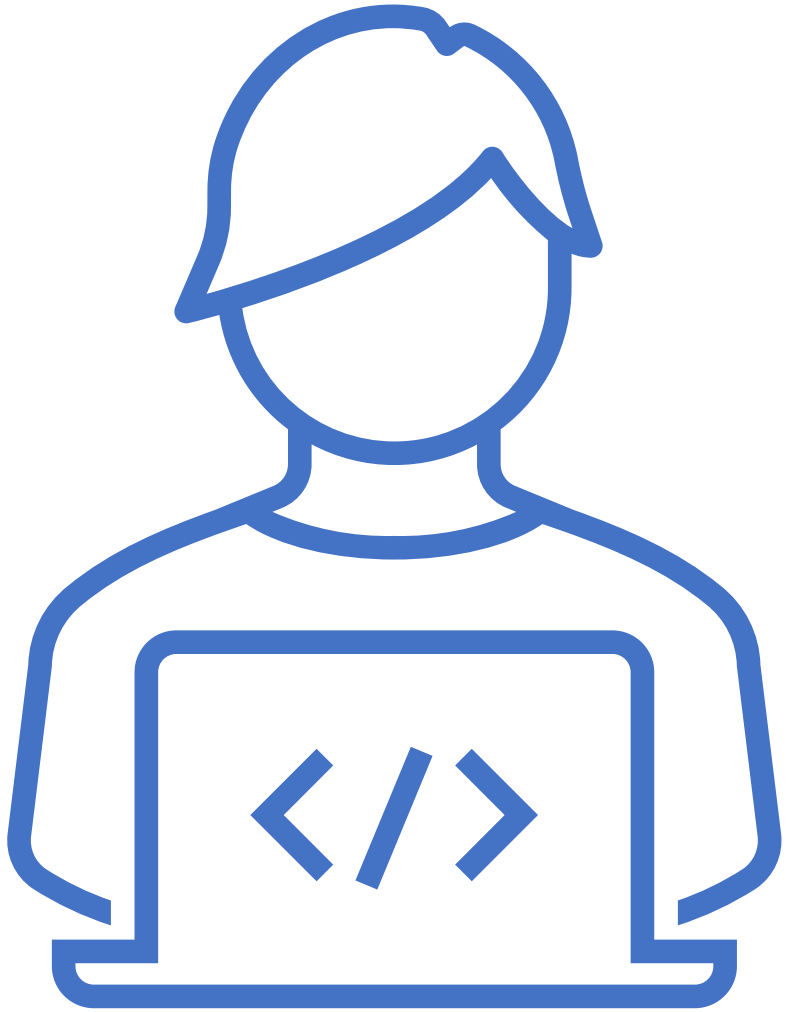
https://azure.microsoft.com/en-us/pricing/details/vpn-gateway/

What protocol does P2S use?

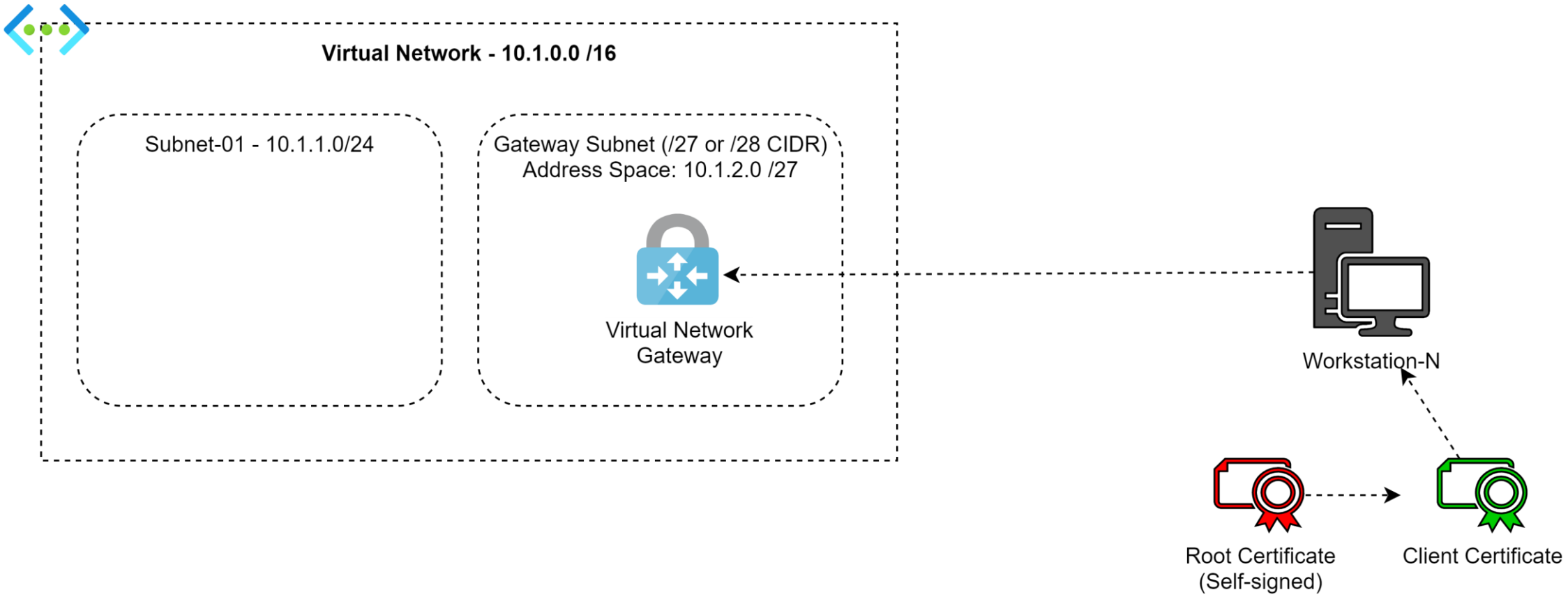
Point-to-site VPN can use one of the following protocols:

1) **Secure Socket Tunneling Protocol (SSTP),** a proprietary SSL-based VPN protocol. An SSL VPN solution can penetrate firewalls, since most firewalls open TCP port 443, which SSL uses. SSTP is only supported on Windows devices. Azure supports all versions of Windows that have SSTP (Windows 7 and later).
2) **IKEv2** VPN, a standards-based IPsec VPN solution. IKEv2 VPN can be used to connect from Mac devices (OSX versions 10.11 and above).
3) If you have a mixed client environment consisting of Windows and Mac devices, configure both SSTP and IKEv2.

Hands-on Labs



Lab: Implementing Point-to-Site VPN connectivity



Steps to establish Point-to-Site Connectivity

1. Create a new Virtual Network
2. Create a Gateway Subnet
3. Create a new Virtual Network Gateway
4. Virtual Network Gateway Setting -> Point-to-site configuration
5. Provide private IP address space (range) for the Clients
6. Generate Root and Client Certificate through PowerShell on the client
7. Upload the root certificate public certificate data on Virtual Network Gateway
8. Generate and install the VPN client configuration package on local system

Point-to-Site Azure certificate authentication connections

Point-to-Site native Azure certificate authentication connections use the following items, which you configure in this exercise:

1. A **RouteBased VPN gateway**.
2. The **Public key (.cer file)** for a root certificate, which is uploaded to Azure. Once the certificate is uploaded, it is considered a trusted certificate and is used for authentication.
3. A **Client certificate** that is generated from the root certificate. The client certificate installed on each client computer that will connect to the VNet. This certificate is used for client authentication.
4. A **VPN client configuration**: The VPN client configuration files contain the necessary information for the client to connect to the VNet. The files configure the existing VPN client that is native to the operating system. Each client that connects must be configured using the settings in the configuration files

How are P2S VPN clients authenticated?

Before Azure accepts a P2S VPN connection, the user has to be authenticated first. There are two mechanisms that Azure offers to authenticate a connecting user:

1. **Authenticate using native Azure certificate authentication**

   When using the native Azure certificate authentication, a client certificate that is present on the device is used to authenticate the connecting user. Client certificates are generated from a trusted root certificate and then installed on each client computer

2. **Authenticate using Active Directory (AD) Domain Server**

   AD Domain authentication allows users to connect to Azure using their organization domain credentials. It requires a RADIUS server that integrates with the AD server. Organizations can also leverage their existing RADIUS deployment

Obtain the .cer file for the Root Certificate

Enterprise certificate: If you are using an enterprise solution, you can use your existing certificate chain. Obtain the .cer file for the root certificate that you want to use.

Self-signed root certificate: If you aren't using an enterprise certificate solution, you need to create a self-signed root certificate

Create Self Signed Certificate using PowerShell

ROOT CERTIFICATE

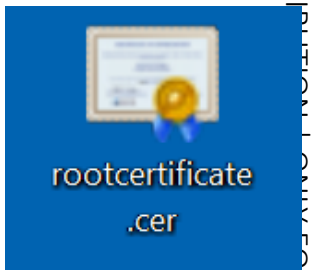
```
$cert = New-SelfSignedCertificate -Type Custom -KeySpec Signature `
-Subject "CN=NovaVPNRoot" -KeyExportPolicy Exportable `
-HashAlgorithm sha256 -KeyLength 2048 `
-CertStoreLocation "Cert:\CurrentUser\My" -KeyUsageProperty Sign -KeyUsage CertSign
```

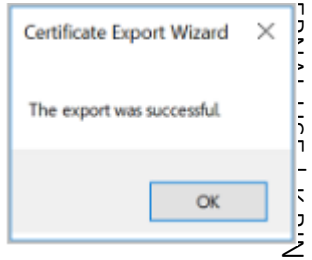
CLIENT CERTIFICATE

```
New-SelfSignedCertificate -Type Custom -KeySpec Signature `
-Subject "CN=P2SChildCert" -KeyExportPolicy Exportable `
-HashAlgorithm sha256 -KeyLength 2048 `
-CertStoreLocation "Cert:\CurrentUser\My" `
-Signer $cert -TextExtension @("2.5.29.37={text}1.3.6.1.5.5.7.3.2")
```

Export the root certificate public key (.cer)

1. Open **certificate manager** in windows platform: Start->Run-> then type **certmgr.msc**.
2. To obtain a .cer file from the certificate, open Manage user certificates. Locate the self-signed root certificate, typically in **'Certificates - Current User\Personal\Certificates'**, and right-click. Click All Tasks, and then click Export. This opens the Certificate Export Wizard.
3. In the Wizard, click **Next**. Select **No**, do not export the private key, and then click Next.
4. On the Export File Format page, select **Base-64 encoded X.509 (.CER)**., and then click Next.
5. On the File to Export, Browse to the location to which you want to export the certificate. For File name, name the certificate file. Then, click **Next**
6. Click **Finish** to export the certificate. You see **The export was successful**. Click OK to close the wizard

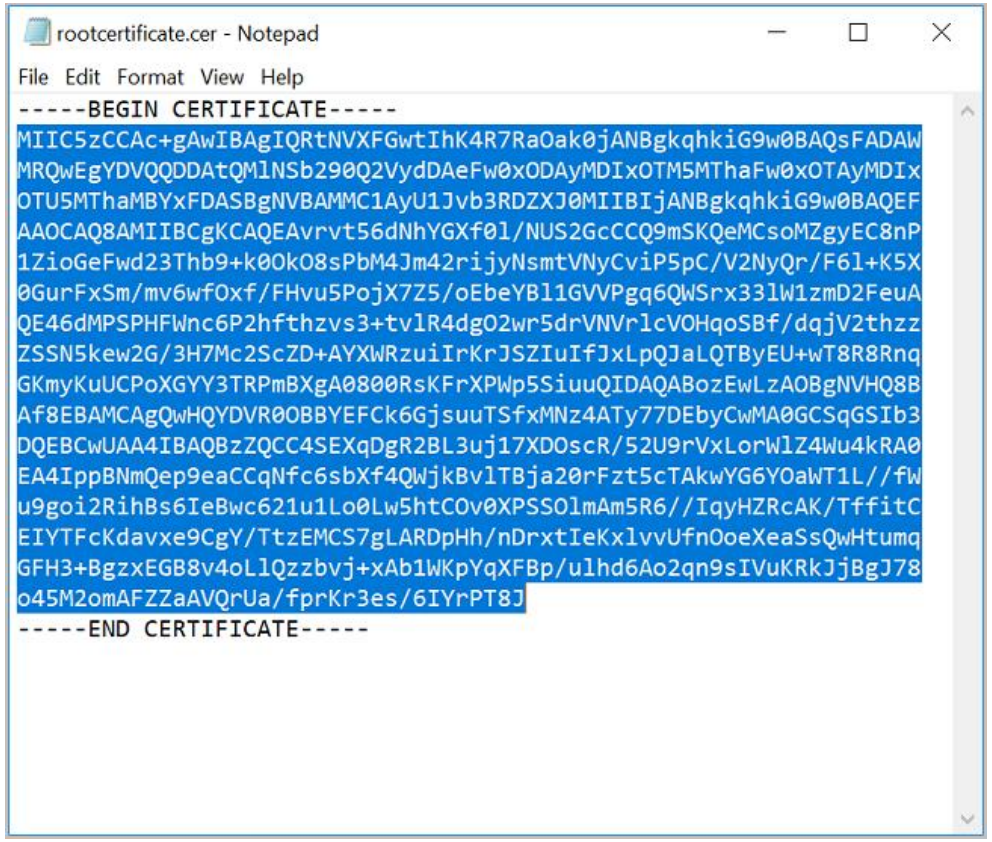




NOVATEC
IT TRAINING & SERVICES

Export the root certificate public key (.cer)

- If you open the exported certificate using Notepad, you see something similar to this example
- The section in blue contains the information that is uploaded to Azure
- If you open your certificate with Notepad and it does not look similar to this, typically this means you did not export it using the Base-64 encoded X.509(.CER) format
- If you want to use a different text editor, understand that some editors can introduce unintended formatting in the background which can create problems when uploaded the text from this certificate to Azure



Export the root certificate public key (.cer)

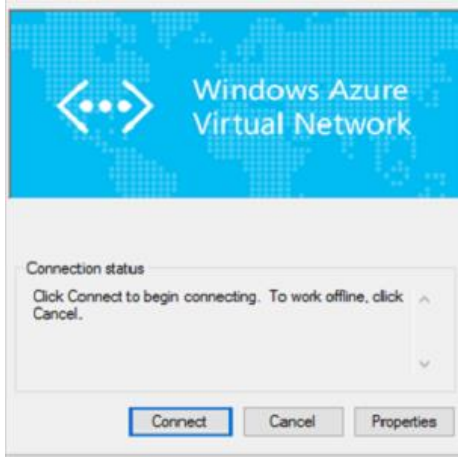
- Paste the certificate data into the **Public Certificate Data** field.
- Name the certificate, and then click **Save**. You can add up to 20 trusted root certificates.
- Click Save at the top of the page to save all of the configuration settings.

Download and install the VPN client configuration package

The VPN client configuration files contain settings to configure devices to connect to a VNet over a P2S connection

Connect to Azure

1. To connect to your VNet, on the client computer, navigate to VPN connections and locate the VPN connection that you created. It is named the same name as your virtual network. Click **Connect**. A pop-up message may appear that refers to using the certificate. Click Continue to use elevated privileges
2. **On the Connection status page**, click **Connect** to start the connection. If you see a Select Certificate screen, verify that the client certificate showing is the one that you want to use to connect. If it is not, use the drop-down arrow to select the correct certificate, and then click OK
3. Your connection is established



To verify your P2S connection

To verify that your VPN connection is active, open an elevated command prompt, and run ipconfig/all.

```
PPP adapter VNet1:
   Connection-specific DNS Suffix .:
   Description.....................: VNet1
   Physical Address................:
   DHCP Enabled....................: No
   Autoconfiguration Enabled.......: Yes
   IPv4 Address....................: 172.16.201.3(Preferred)
   Subnet Mask.....................: 255.255.255.255
   Default Gateway.................:
   NetBIOS over Tcpip..............: Enabled
```

Export the client certificate

When you generate a client certificate, it's automatically installed on the computer that you used to generate it. If you want to install the client certificate on another client computer, you need to export the client certificate that you generated.

1. To export a client certificate, open **Manage user certificates**. The client certificates that you generated are, by default, located in 'Certificates - Current User\Personal\Certificates'. Right-click the client certificate that you want to export, click all tasks, and then click Export to open the Certificate Export Wizard
2. In the Certificate Export Wizard, click **Next** to continue
3. Select **Yes**, export the private key, and then click **Next**.
4. On the Export File Format page, leave the defaults selected. Make sure that Include all certificates in the certification path if possible is selected. This setting additionally exports the root certificate information that is required for successful client authentication. Without it, client authentication fails because the client doesn't have the trusted root certificate. Then, click **Next**

Export the client certificate

5. On the Security page, you must protect the private key. If you select to use a password, make sure to record or remember the password that you set for this certificate. Then, click **Next**
6. On the File to Export, Browse to the location to which you want to export the certificate. For File name, name the certificate file. Then, click **Next**
7. Click Finish to export the certificate

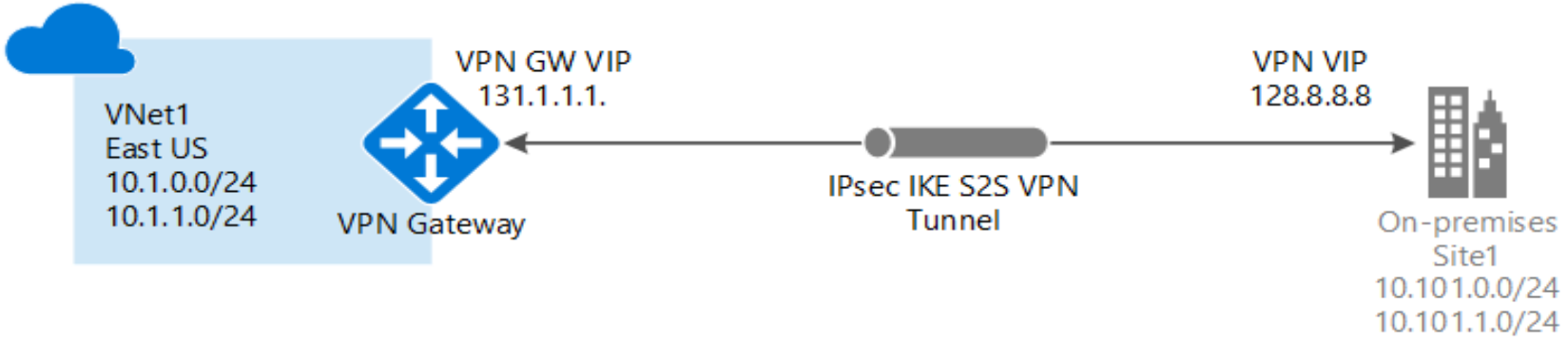
Install certificate - Windows

If you want to create a P2S connection from a client computer other than the one you used to generate the client certificates, you need to install a client certificate. When installing a client certificate, you need the password that was created when the client certificate was exported.

1. Locate and copy the .pfx file to the client computer. On the client computer, double-click the .pfx file to install. Leave the Store Location as Current User, and then click Next.
2. On the File to import page, don't make any changes. Click Next.
3. On the Private key protection page, input the password for the certificate, or verify that the security principal is correct, then click Next.
4. On the Certificate Store page, leave the default location, and then click Next.
5. Click Finish. On the Security Warning for the certificate installation, click Yes. You can feel comfortable clicking **'Yes'** because you generated the certificate. The certificate is now successfully imported.

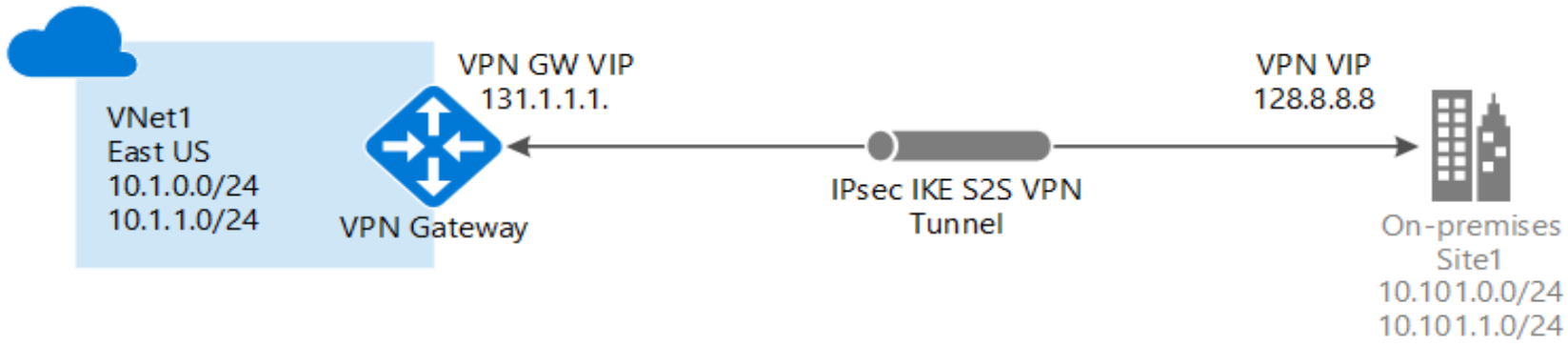
Site to Site VPN Connectivity

- A **Site-to-Site VPN gateway connection** is used to connect your on-premises network to an Azure virtual network over an IPsec/IKE (IKEv1 or IKEv2) VPN tunnel
- This type of connection requires a VPN device located on-premises that has an externally facing public IP address assigned to it
- Make sure you have a compatible VPN device and someone who is able to configure it.
- Verify that you have an externally facing public IPv4 address for your VPN device. This IP address cannot be located behind a NAT



Steps to establish Site to Site VPN Connectivity

1. Create Virtual Network
2. Create Gateway Subnet
3. Create Virtual Network Gateway
4. Create Local Network Gateway
5. Create Site-to-Site VPN

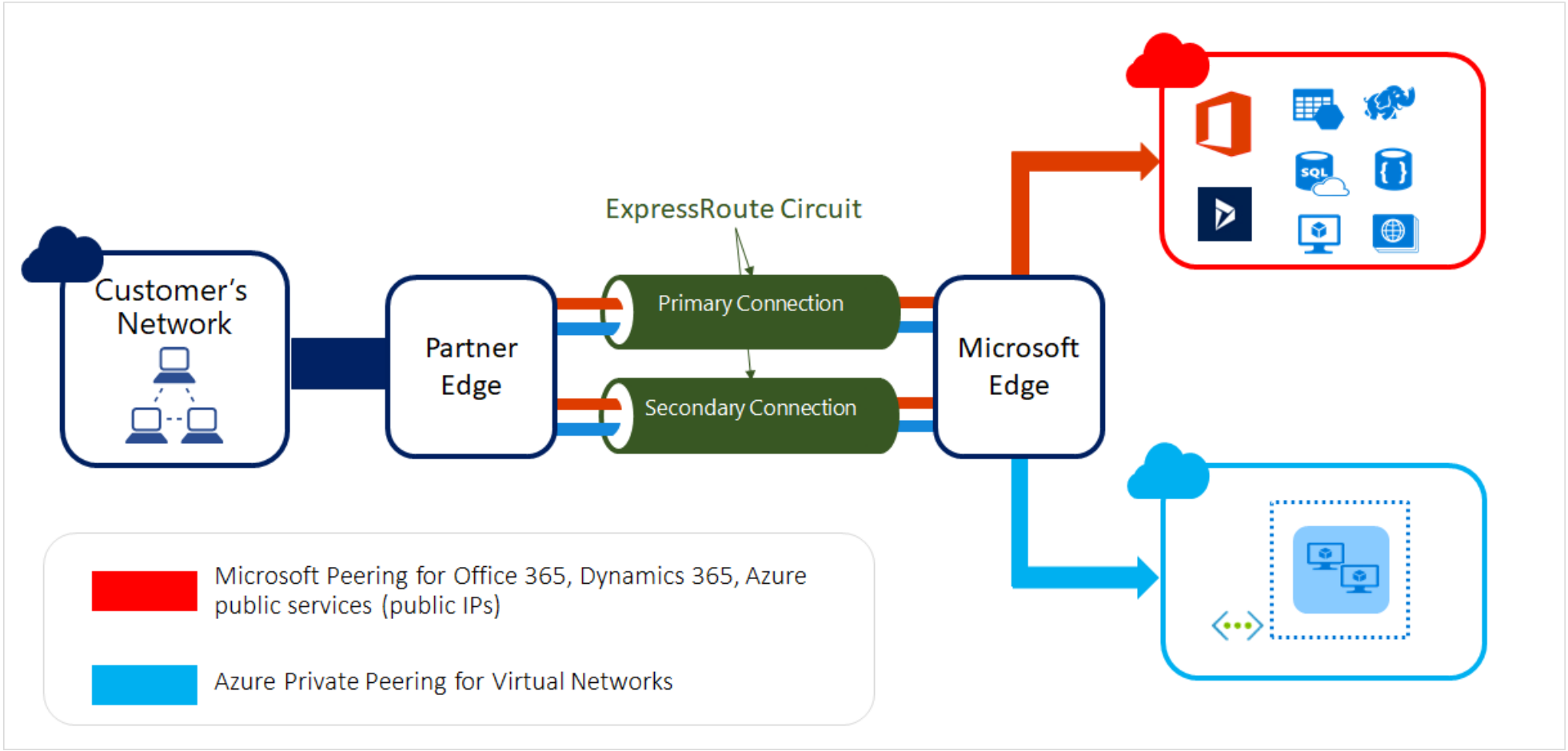


Azure ExpressRoute Connectivity

- ExpressRoute lets you extend your on-premises networks into the Microsoft cloud over a **private connection** with the help of a **connectivity provider**
- Connectivity can be from an any-to-any (IP VPN) network, a point-to-point Ethernet network, or a virtual cross-connection through a connectivity provider at a colocation facility
- **ExpressRoute connections don't go over the public Internet**
- This allows ExpressRoute connections to offer more reliability, faster speeds, consistent latencies, and higher security than typical connections over the Internet

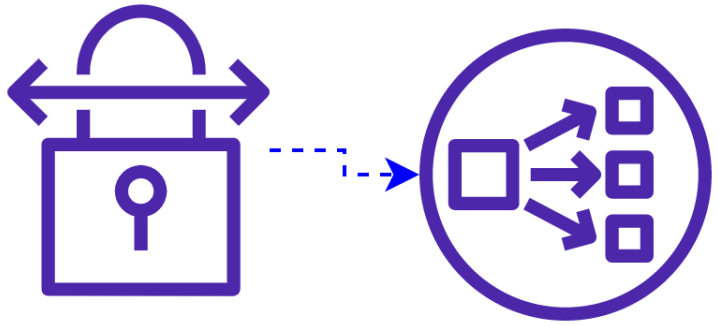
Ref: https://docs.microsoft.com/en-in/azure/expressroute/expressroute-introduction

Azure ExpressRoute Connectivity



Azure ExpressRoute Connectivity: Advantages

- ✓ Layer 3 connectivity between your on-premises network and the Microsoft Cloud through a connectivity provider Connectivity can be from an any-to-any (IPVPN) network, a point-to-point Ethernet connection, or through a virtual cross-connection via an Ethernet exchange.
- ✓ Connectivity to Microsoft cloud services across all regions in the geopolitical region
- ✓ Global connectivity to Microsoft services across all regions with the ExpressRoute premium add-on
- ✓ Dynamic routing between your network and Microsoft via BGP
- ✓ Built-in redundancy in every peering location for higher reliability



Use Case : VPN Connection and Load Balancer

Azure DNS

Domain Name System (DNS) Primer

- <content here>

Overview of Azure DNS

- Azure DNS is a hosting service for DNS domains that provides name resolution by using Microsoft Azure infrastructure
- By hosting your domains in Azure, you can manage your DNS records by using the same credentials, APIs, tools, and billing as your other Azure services
- You can't use Azure DNS to buy a domain name
- For an annual fee, you can buy a domain name by using App Service domains or a third-party domain name registrar
- Your domains then can be hosted in Azure DNS for record management

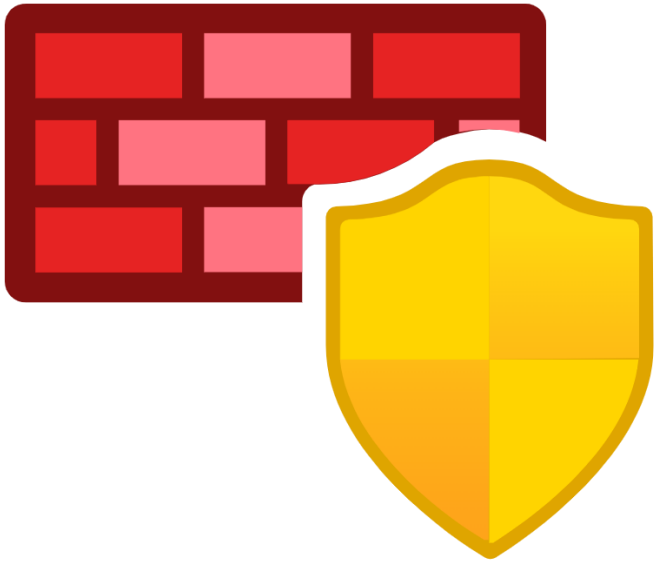
Ref: https://docs.microsoft.com/en-us/azure/dns/dns-overview

What is Azure Private DNS?

- The Domain Name System, or DNS, is responsible for translating (or resolving) a service name to an IP address
- Azure DNS is a hosting service for domains and provides naming resolution using the Microsoft Azure infrastructure
- Azure DNS not only supports internet-facing DNS domains, but it also supports private DNS zones
- Azure Private DNS provides a reliable and secure DNS service for your virtual network
- Azure Private DNS manages and resolves domain names in the virtual network without the need to configure a custom DNS solution

What is Azure Public DNS?

- You can configure Azure DNS to resolve host names in your public domain
- For example, if you purchased the ***novatecpune.com*** domain name from a domain name registrar, you could configure Azure DNS to host the ***novatecpune.com*** domain and resolve ***novatecpune.com*** to the IP address of your web server or web app



Azure Firewall

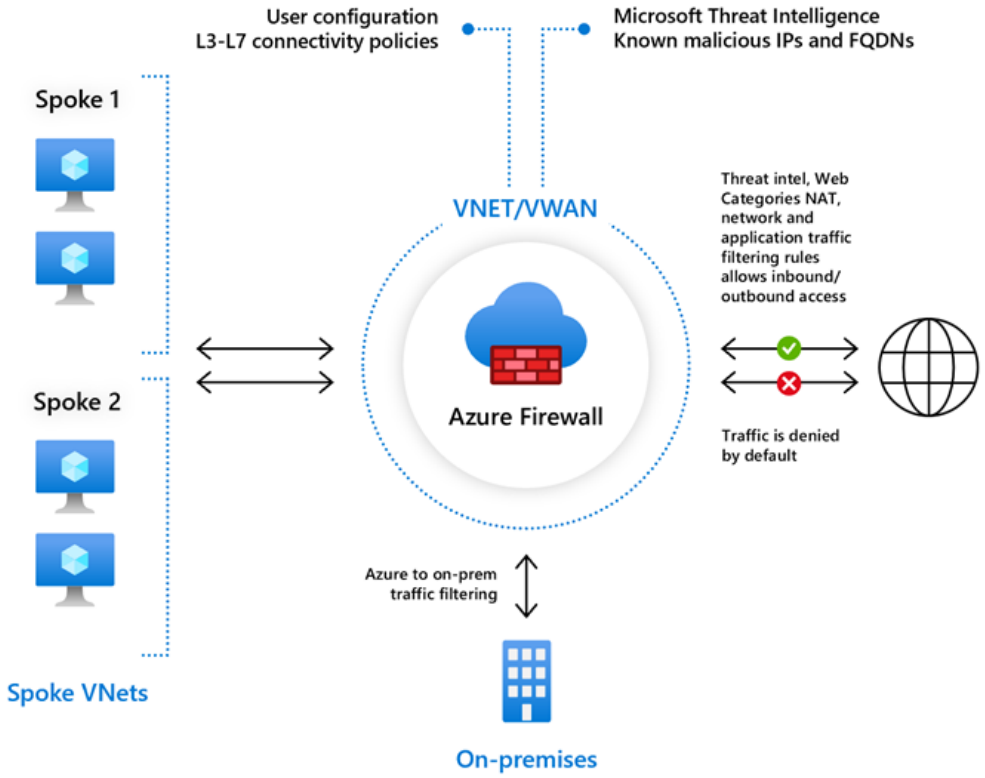
Overview of Azure Firewall

- Azure Firewall is a cloud-native and intelligent network firewall security service that provides the best threat protection for your cloud workloads running in Azure
- It's a fully stateful, firewall as a service with built-in high availability and unrestricted cloud scalability
- It provides both east-west and north-south traffic inspection
- Azure Firewall is offered in two SKUs:
  1. Standard
  2. Premium

Ref: https://docs.microsoft.com/en-us/azure/firewall/overview

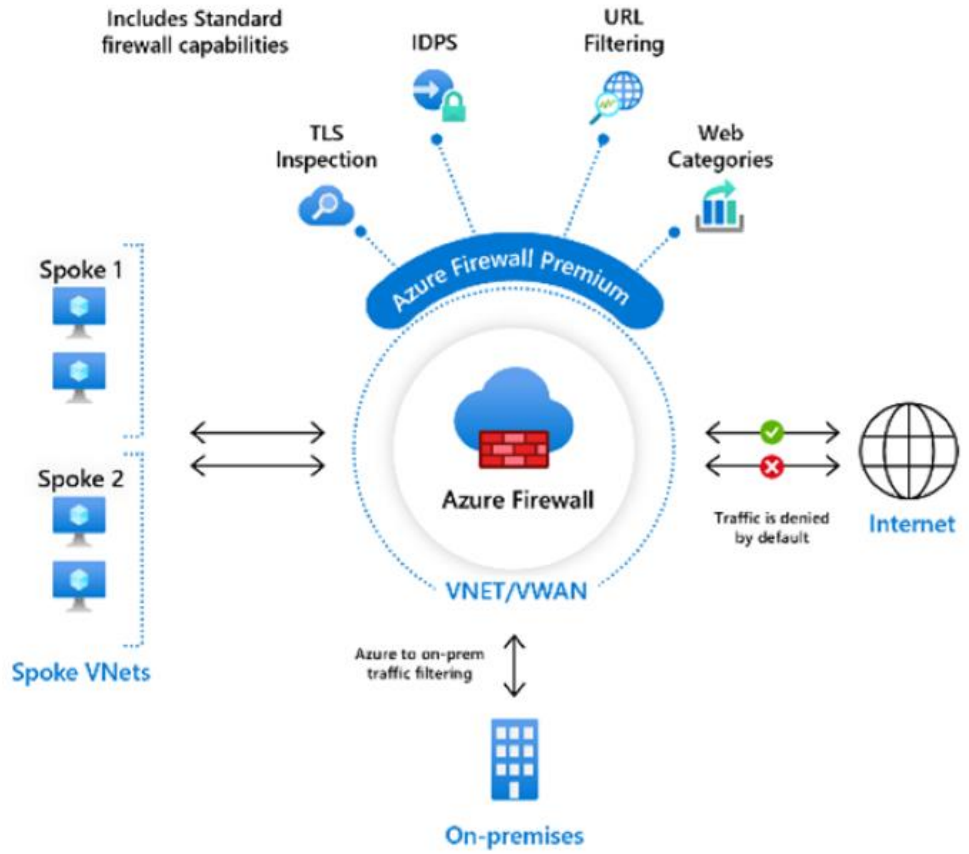
Azure Firewall Standard

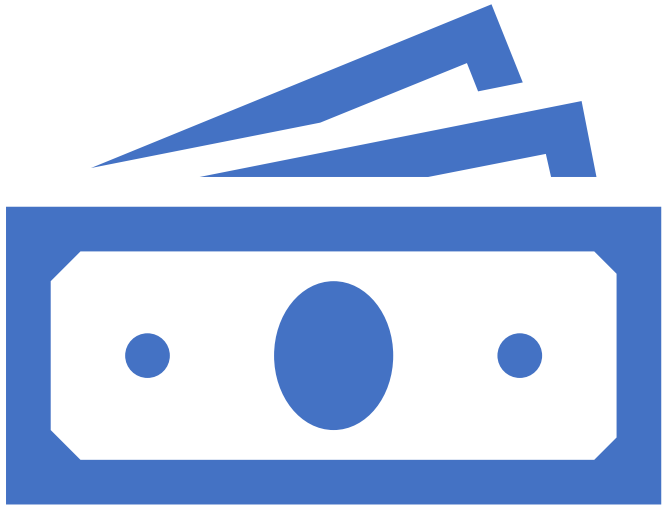
- Azure Firewall Standard provides L3-L7 filtering and threat intelligence feeds directly from Microsoft Cyber Security
- Threat intelligence-based filtering can alert and deny traffic from/to known malicious IP addresses and domains which are updated in real time to protect against new and emerging attacks



Azure Firewall Premium

- Azure Firewall Premium provides advanced capabilities include signature-based IDPS to allow rapid detection of attacks by looking for specific patterns
- These patterns can include byte sequences in network traffic, or known malicious instruction sequences used by malware
- More than 58,000 signatures in 58 categories





Azure Virtual Network: Pricing

Azure Virtual Network – Pricing

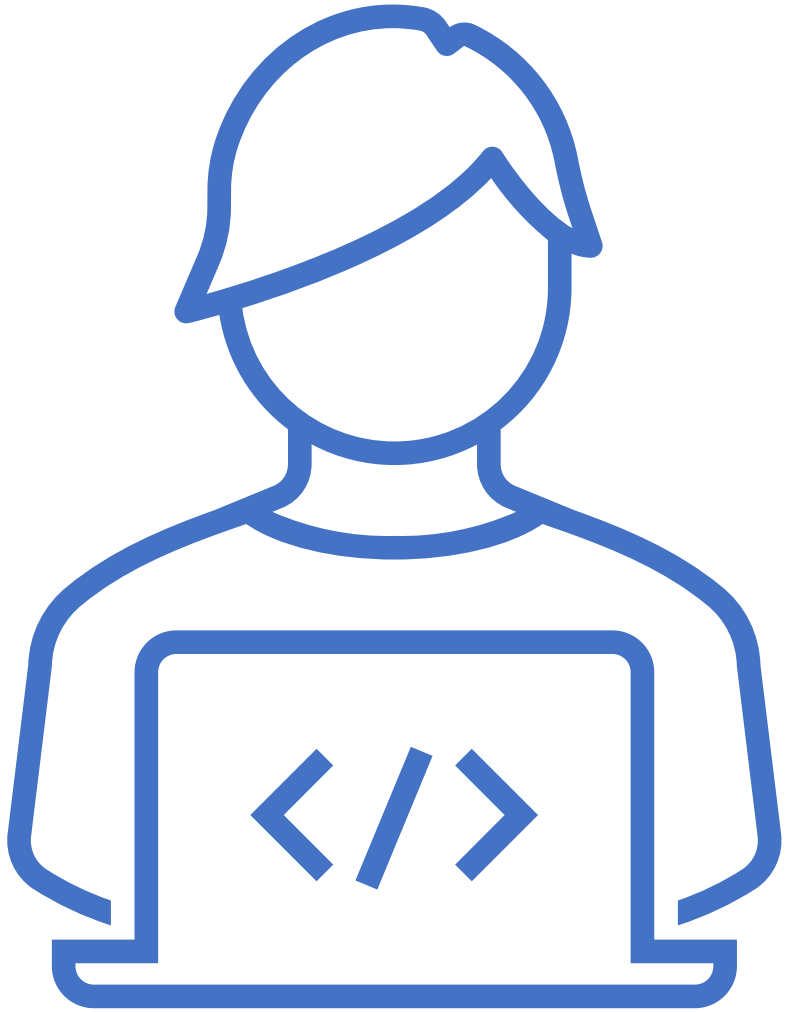
- You are charged for the **public IP** address and **reserved IP** address inside your VNet
- You are charged for the **ingress** and **egress** data of VNet Peering
- You are charged for the **NAT gateway resource hours** and **data processed (per GB)**

Network Watcher

Overview of Network Watcher service

Connection Monitoring	Next Hop	Connection Troubleshoot	IP Flow Verification	NSG Diagnostics & Flow Logs
• Check the network connectivity between machines. • These machines can be Azure or on-premise environments	• You can see the next route for a packet of data	• Check the connection from a VM to VM, FQDN, and URI or IPv4	• This feature can be used to check if a data packet is allowed or denied to or from a VM.	• Provides detailed information about network security configuration for debugging • Helps to provide the visibility into users and application activity in cloud network

Hands-on Labs



Lab: Demonstrating working of Network Watcher service in different use-cases

1) Troubleshooting connections using Network Watcher
2) Connection monitoring using Network Watcher
3) Verify IP Flow using Network Watcher
4) Finding next hop using Network Watcher
5) Network Security Group Diagnostics using Network Watcher