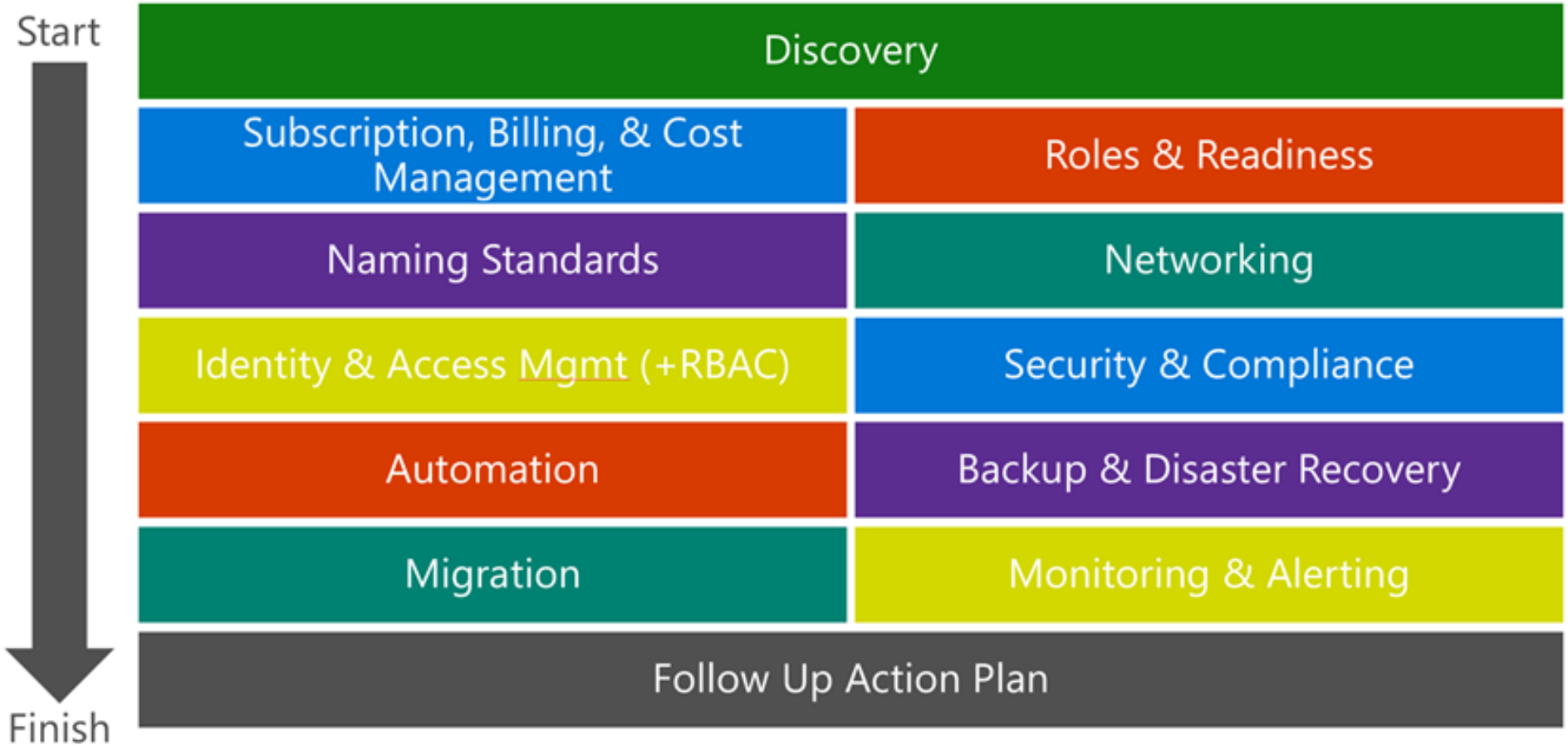
Microsoft

Azure Core Enablement

Networking

Areas of focus

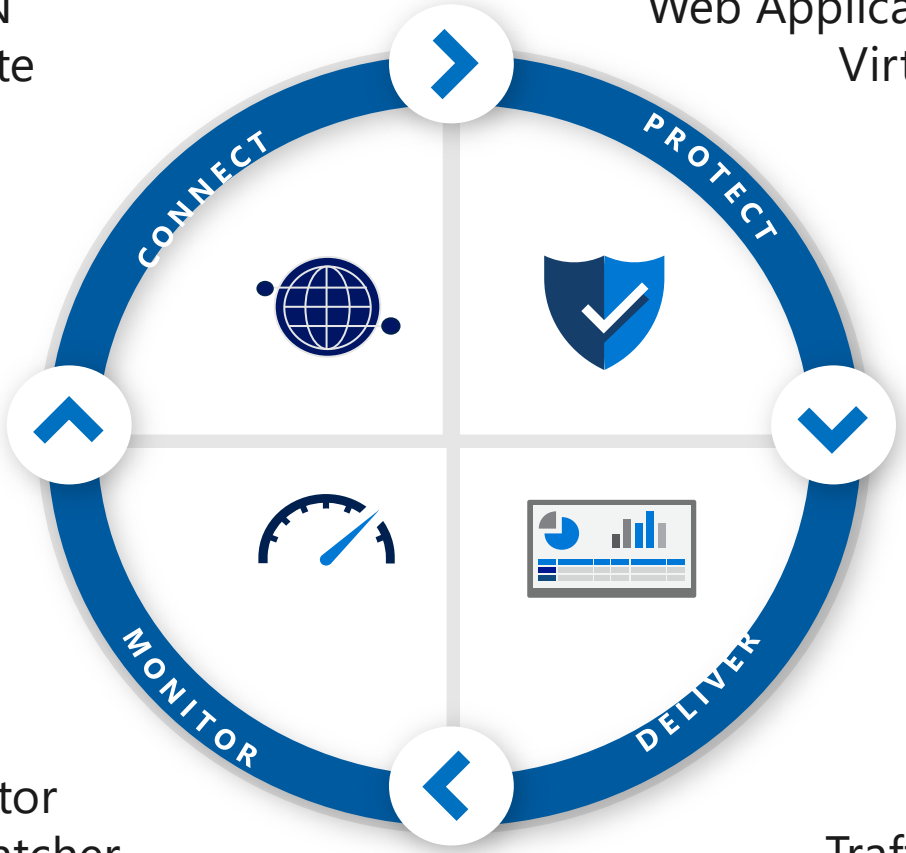


Azure Networking

Rich portfolio of networking services

Virtual Network
Virtual WAN
ExpressRoute
VPN
DNS

DDoS Protection
Web Application Firewall
Virtual Network
Endpoints
Firewall
NSG



Azure Monitor
Network Watcher
ExpressRoute Monitor
Virtual Network TAP

CDN
Front Door
Traffic Manager
Application Gateway
Load Balancer

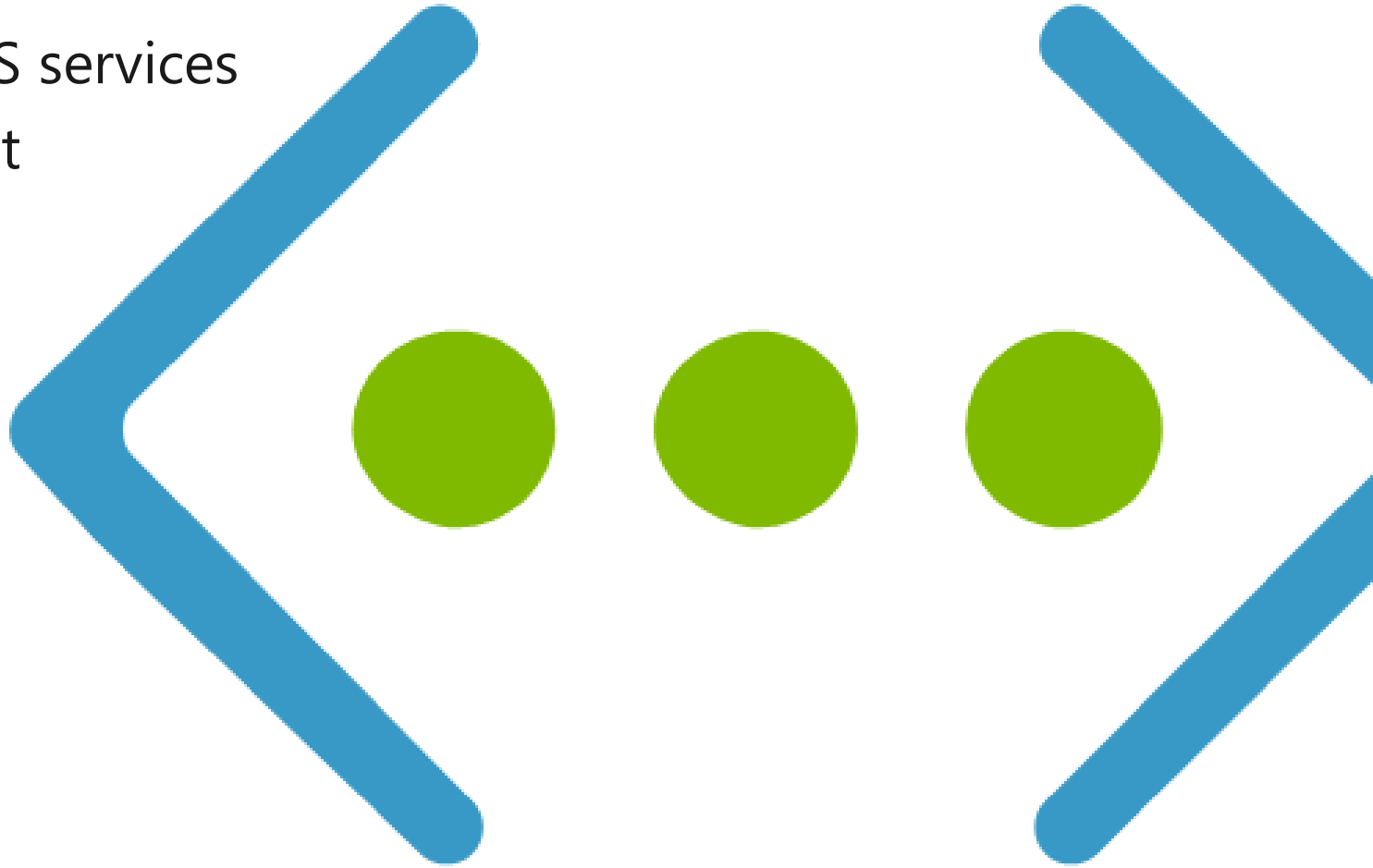
Virtual Networks - VNets

Private, isolated network for customer

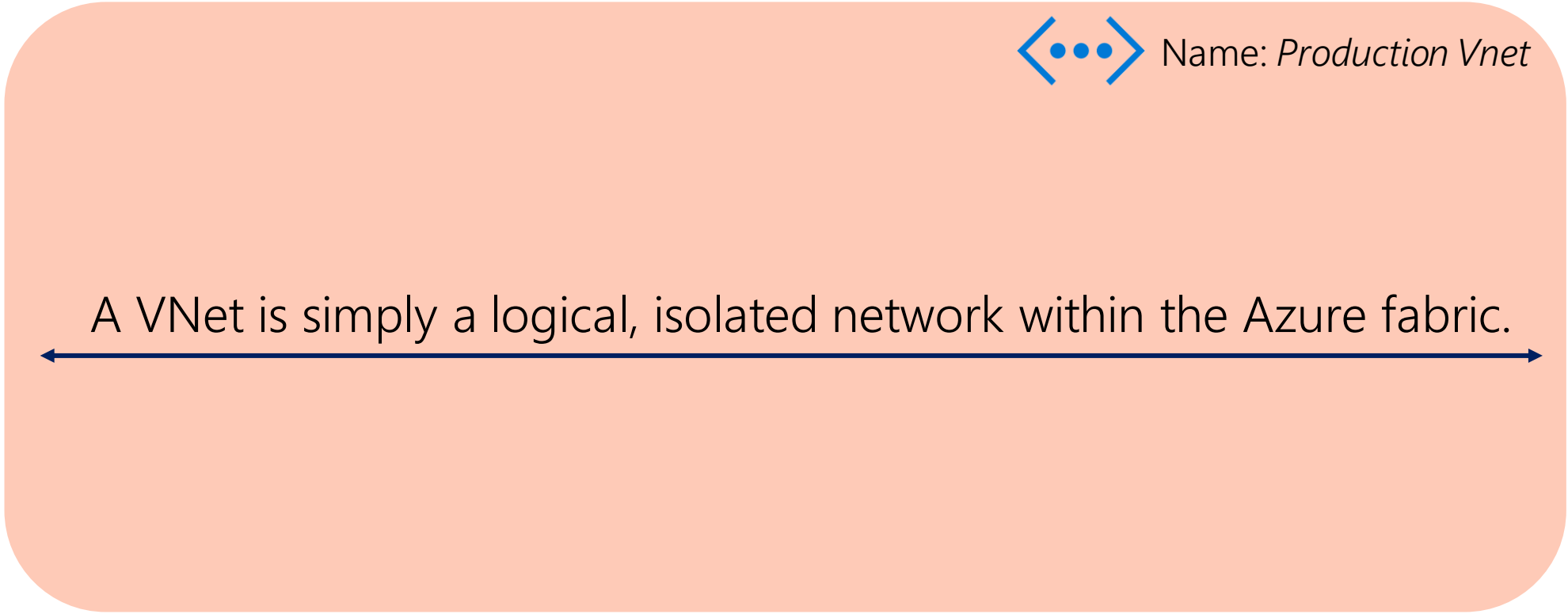
VNet Injection for dedicated PaaS services

Service Endpoints for shared PaaS services

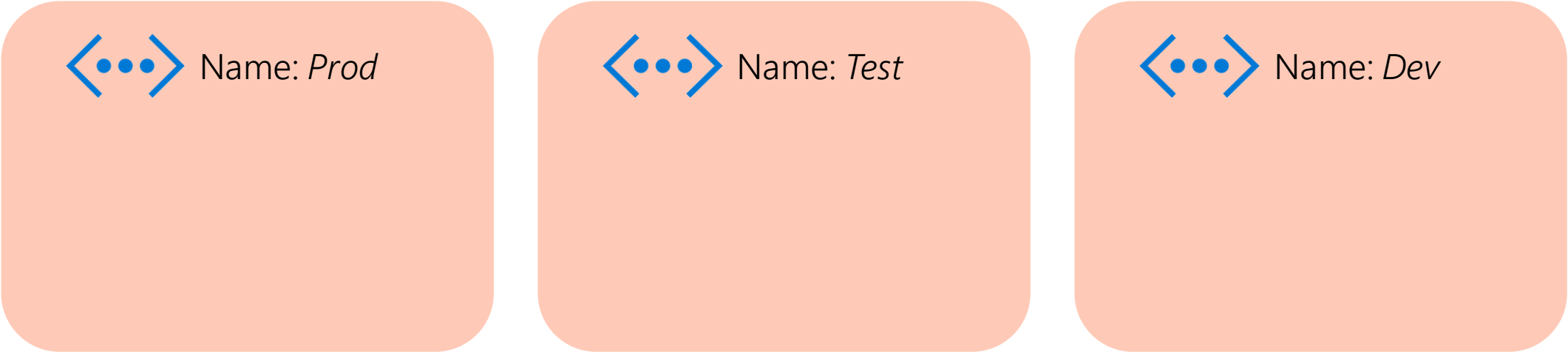
64,000 private IPs (VMs) in a VNet



Azure networking is built around the concept of *Virtual Networks* (VNets)



VNets are (by default) completely isolated from each other.

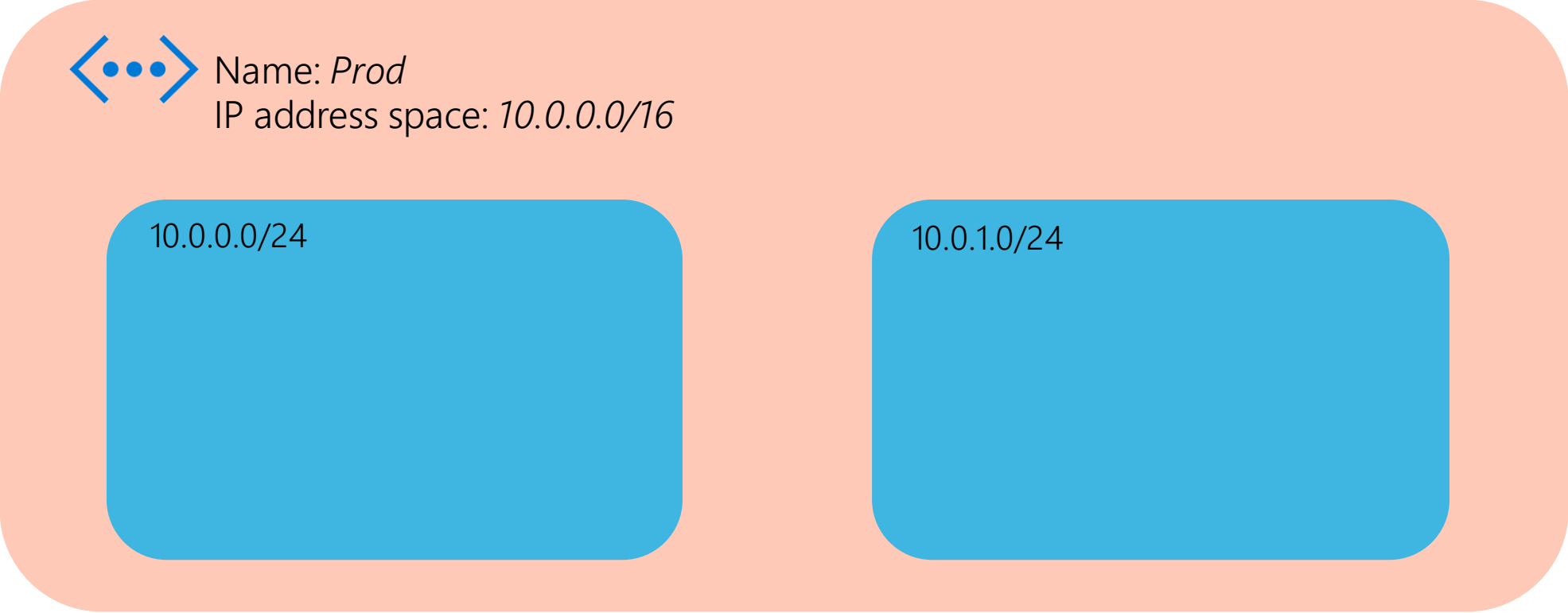


A VNet must be configured with at least one IP address space.

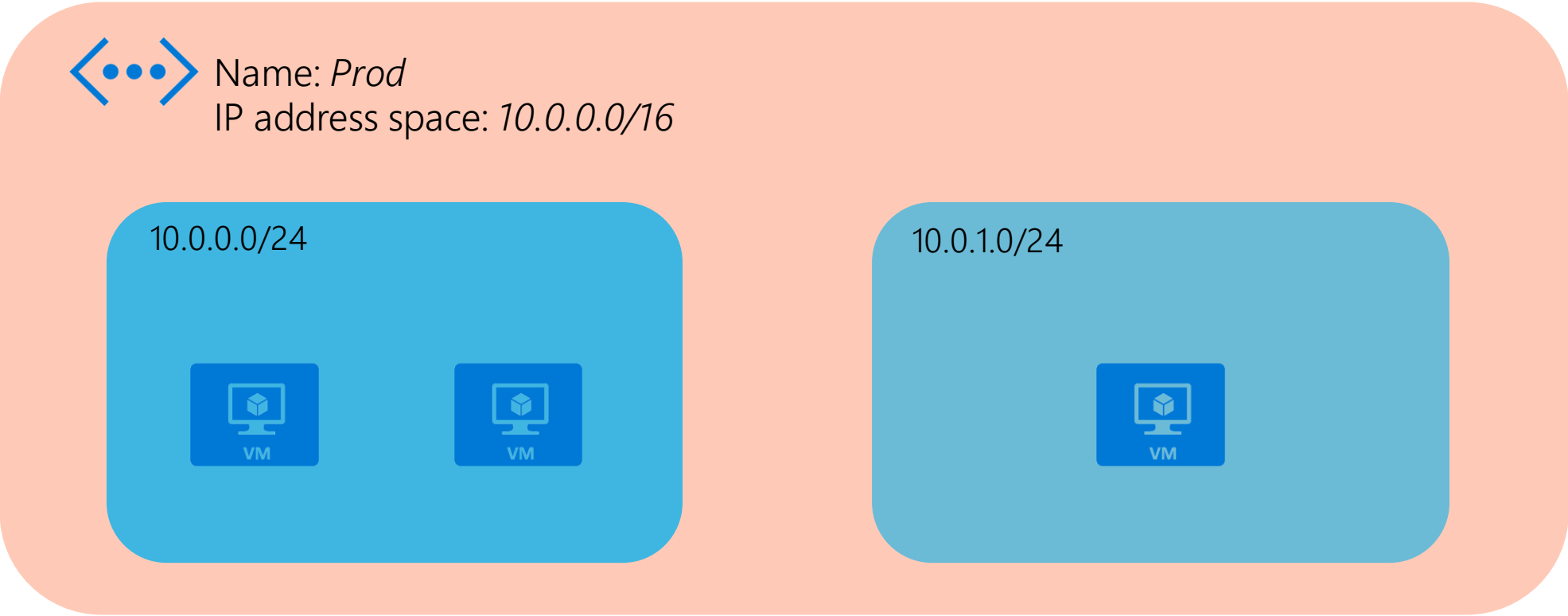
Name: *Prod*
IP address space: *10.0.0.0/16, 192.168.1.0/24*

It is possible for a VNet to have more than one address space assigned to it.

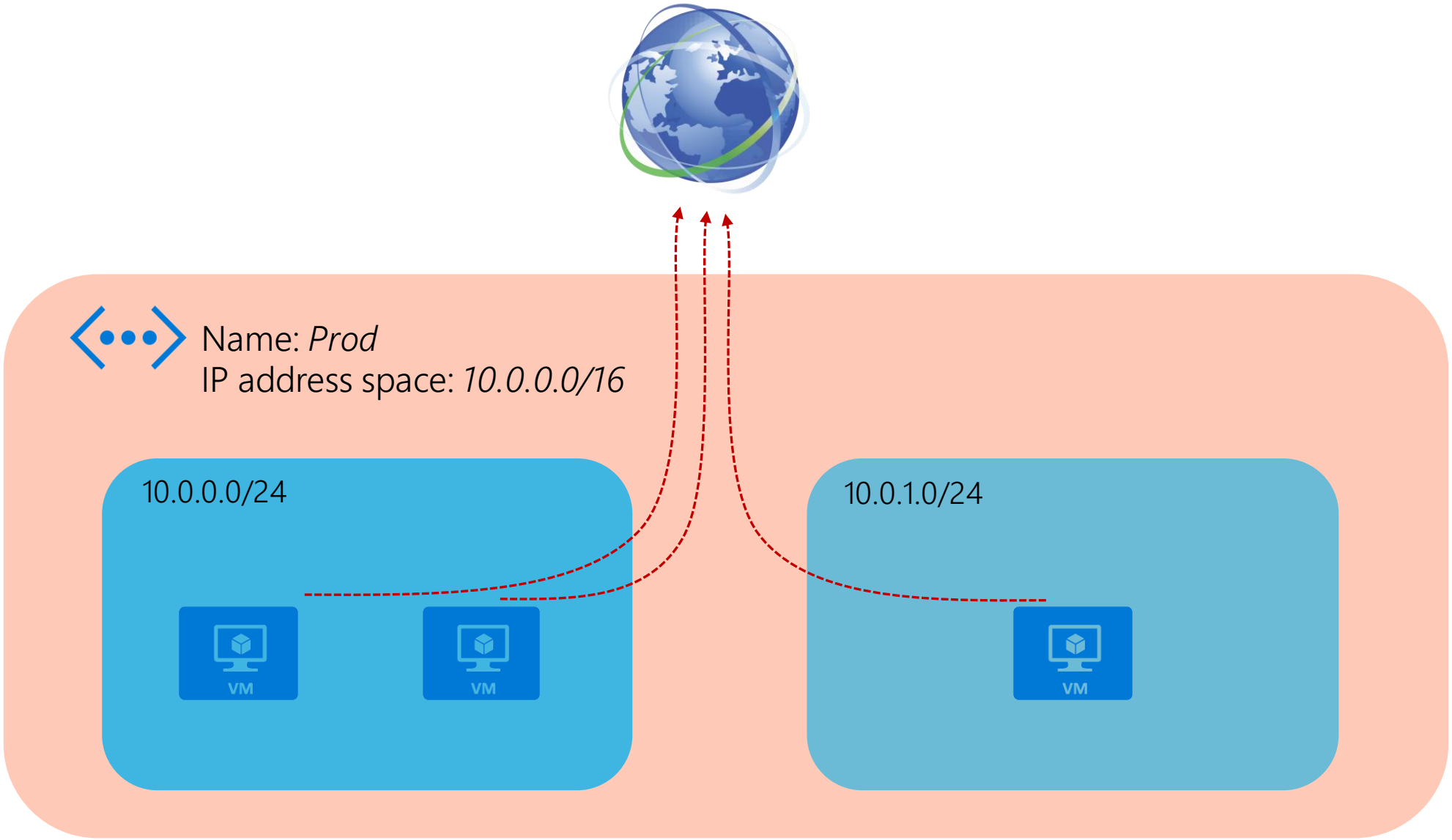
In order to create a Vnet at least one subnet need to exist inside the Vnet, however you can many *subnets*.



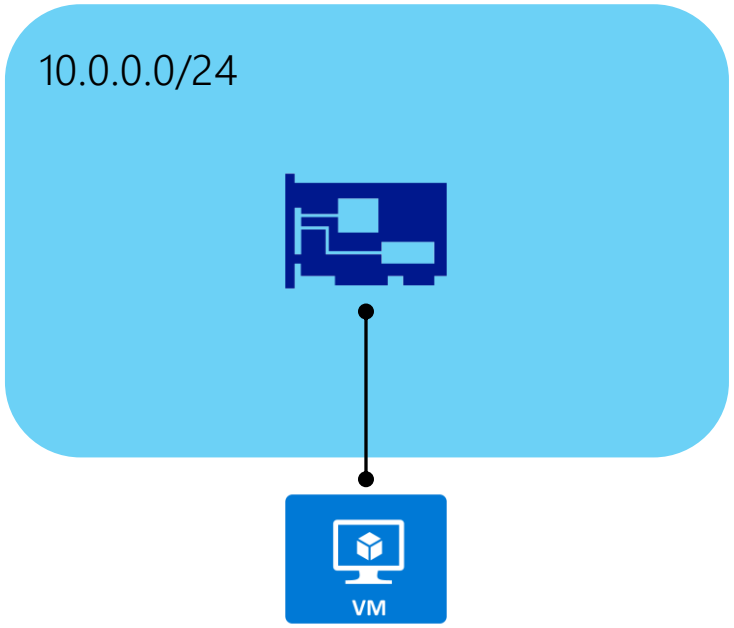
Virtual machines consume IPs from the subnet inside the VNet.



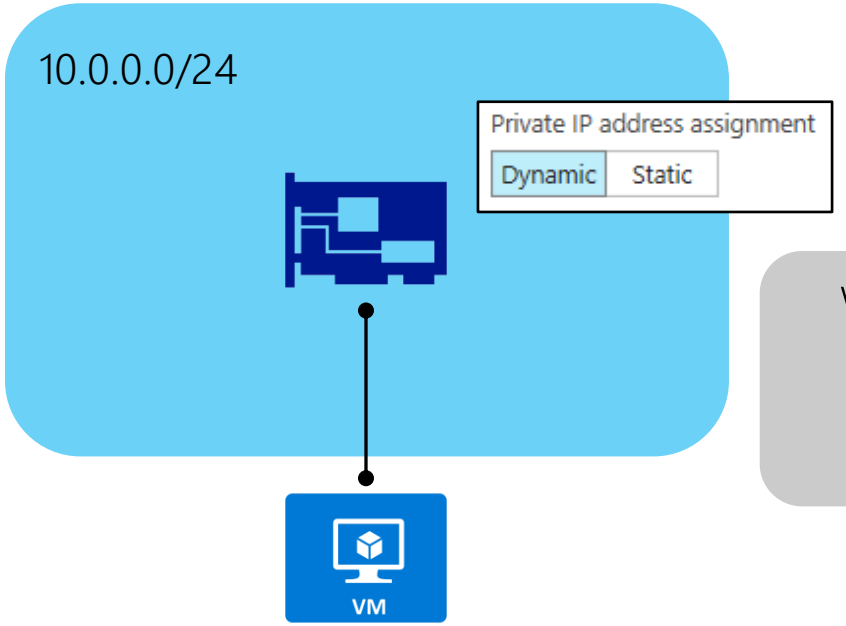
VMs have outbound Internet connectivity *by default*.



Technically, it is the *Network Interface* that connects a VM to a subnet.

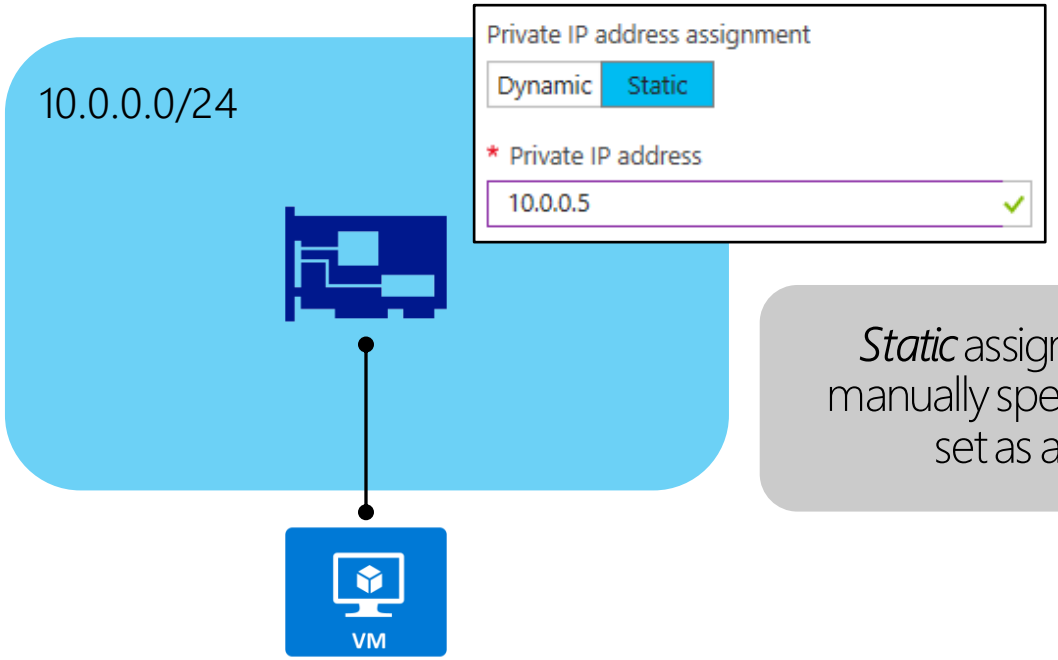


Private IP addresses (i.e. taken from the subnet range) can be allocated either *dynamically* or *statically*.



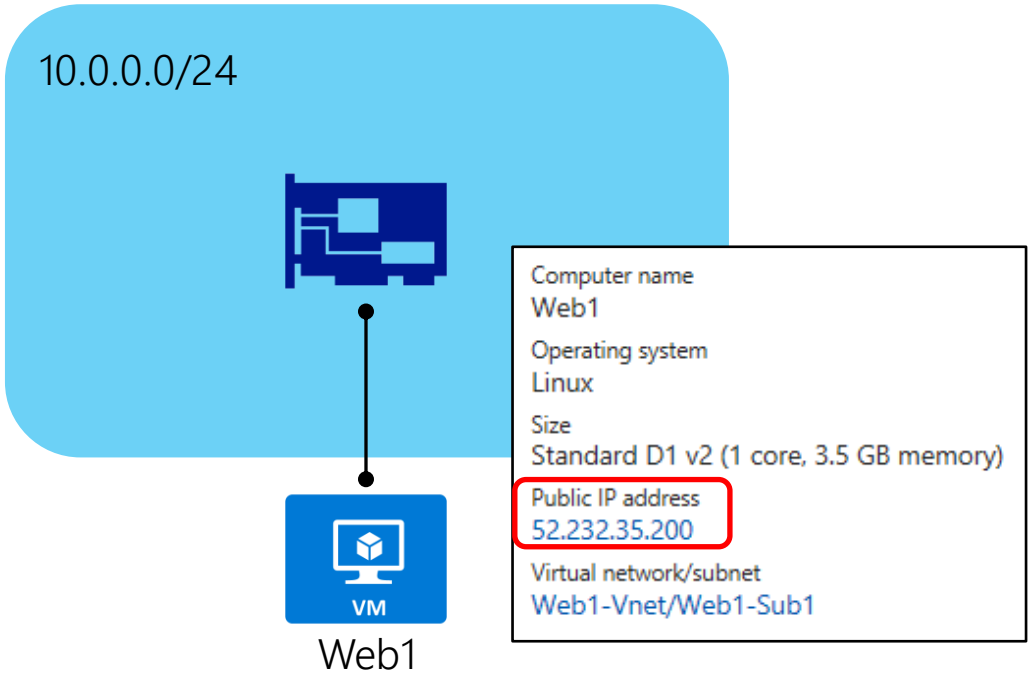
With *dynamic* assignment, addresses are automatically allocated by the DHCP server when the VM starts and **may not remain the same** when the VM reboots.

Private IP addresses (i.e. taken from the subnet range) can be allocated either *dynamically* or *statically*.

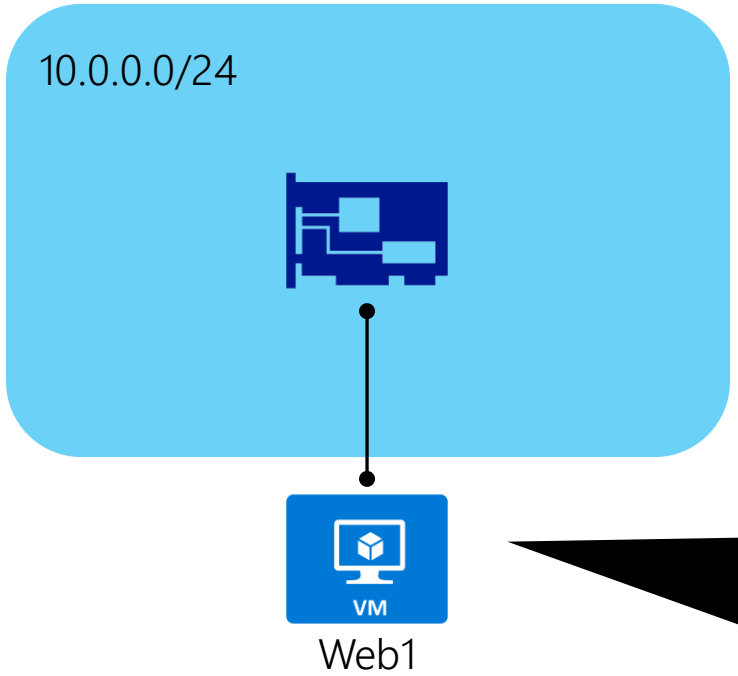


Static assignment means that you can manually specify the address and it will be set as a reservation by DHCP.

A VM can also have a *Public IP* assigned to it – by doing so, the VM will be accessible from the Internet.



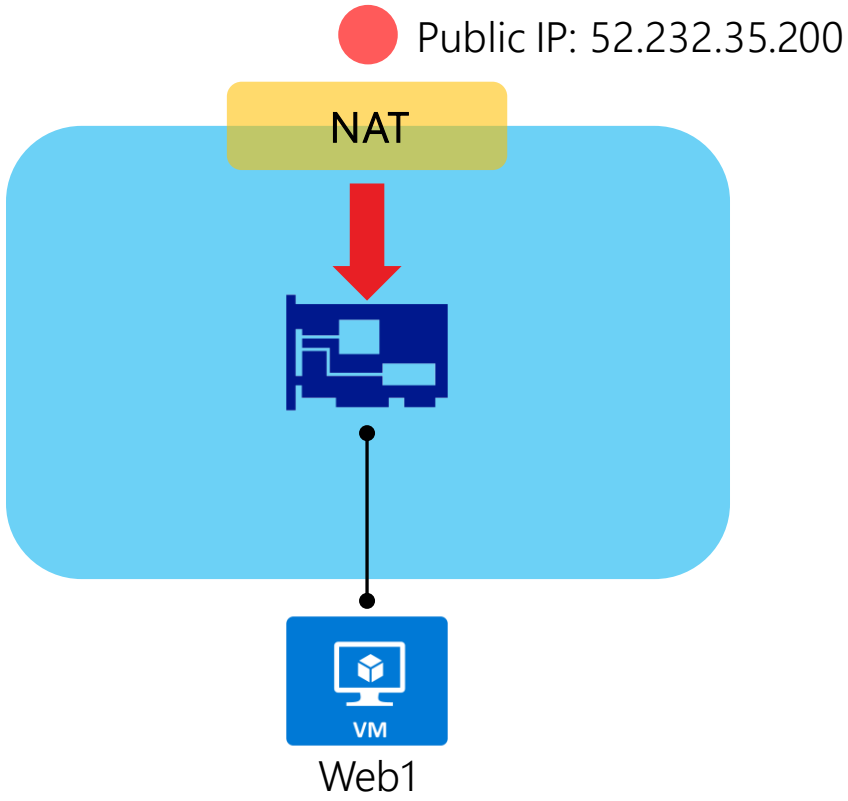
But…when I look at the interfaces on my VM, I don't see my public IP?!



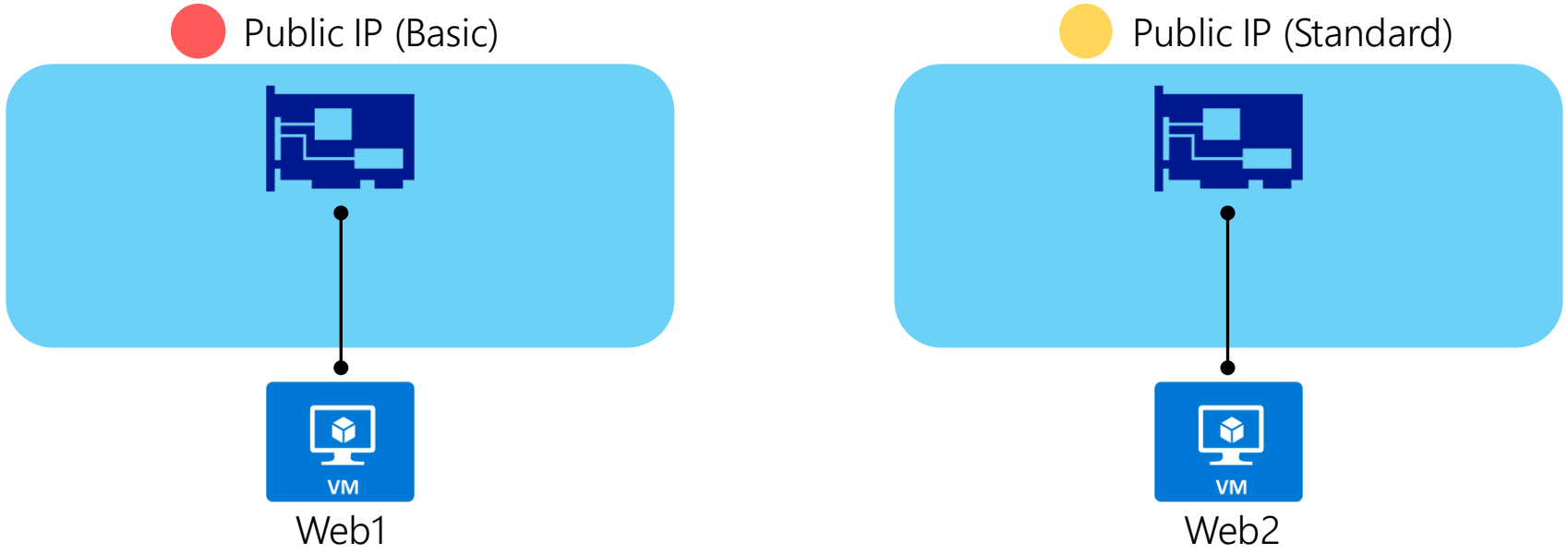

```
adraffe@Web1:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.0.4  netmask 255.255.255.0  broadcast 10.0.0.255
        inet6 fe80::20d:3aff:fe23:940d  prefixlen 64  scopeid 0x20<link>
        ether 00:0d:3a:23:94:0d  txqueuelen 1000  (Ethernet)
        RX packets 1210067  bytes 395242257 (395.2 MB)
        RX errors 0  dropped 17  overruns 0  frame 0
        TX packets 1270939  bytes 218917955 (218.9 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 326  bytes 24172 (24.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 326  bytes 24172 (24.1 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

This is because the public IP actually exists as a NAT (Network Address Translation) entry on the Azure fabric that gets mapped to the VM.

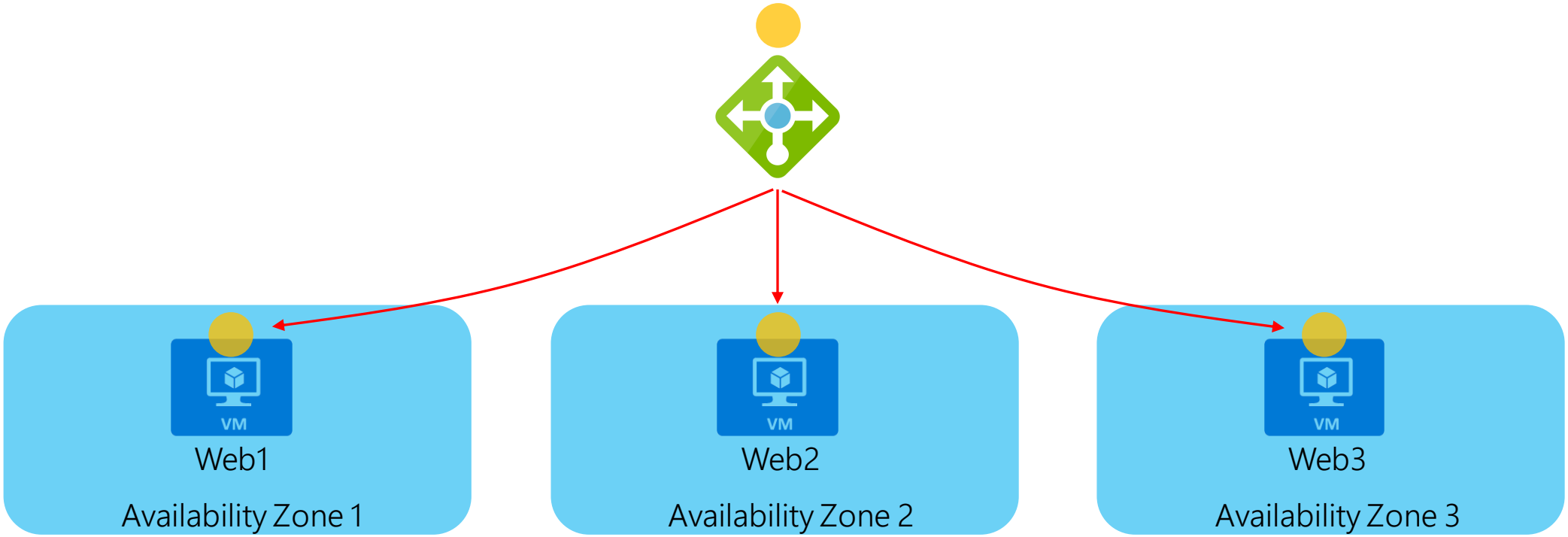


Public IP addresses are available in two “SKUs” – *Basic* or *Standard*.

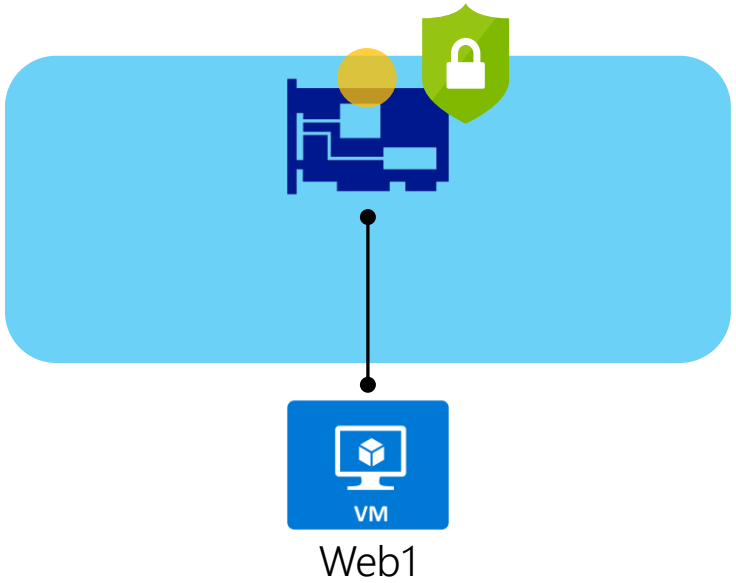


Note: all networking devices must share the same “SKU” level (i.e. Standard IP = Standard LB, etc.)

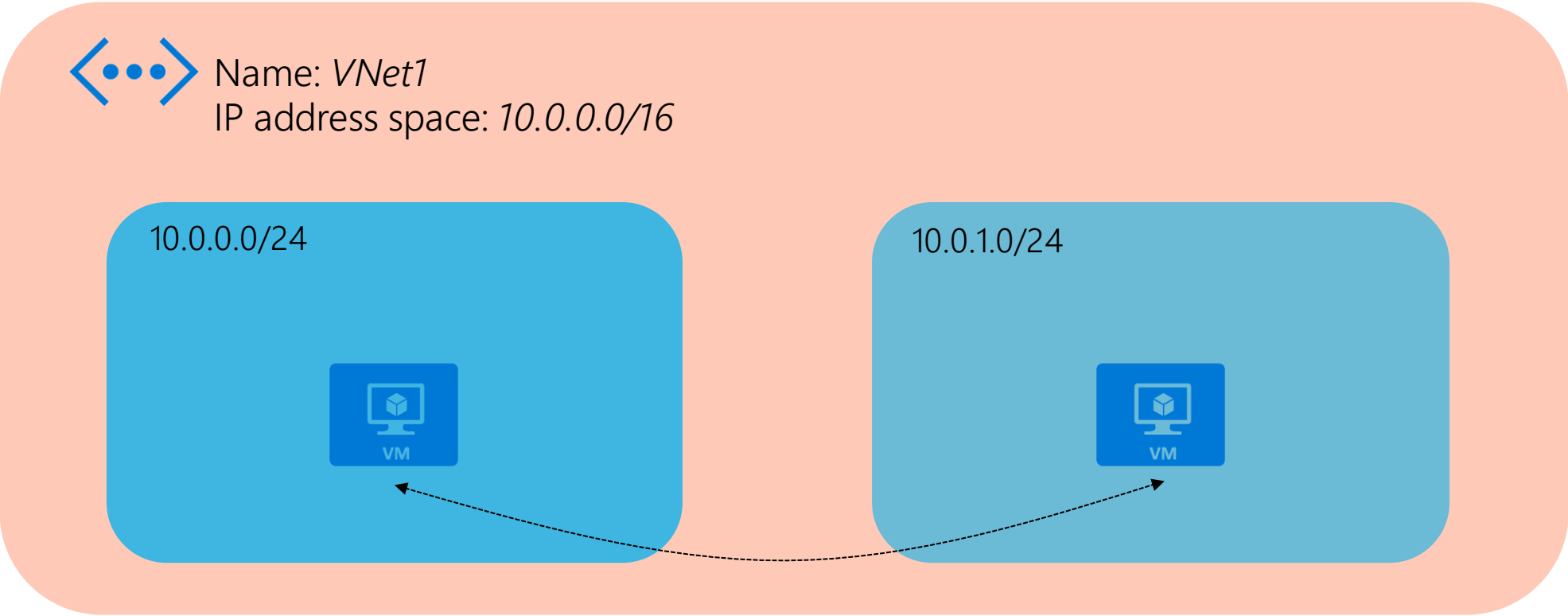
The main difference with Standard Public IP addresses are that they are *zone redundant*.



Caution: If attaching a Public IP to a VM, you *must* apply a Network Security Group or the VM will be unreachable.

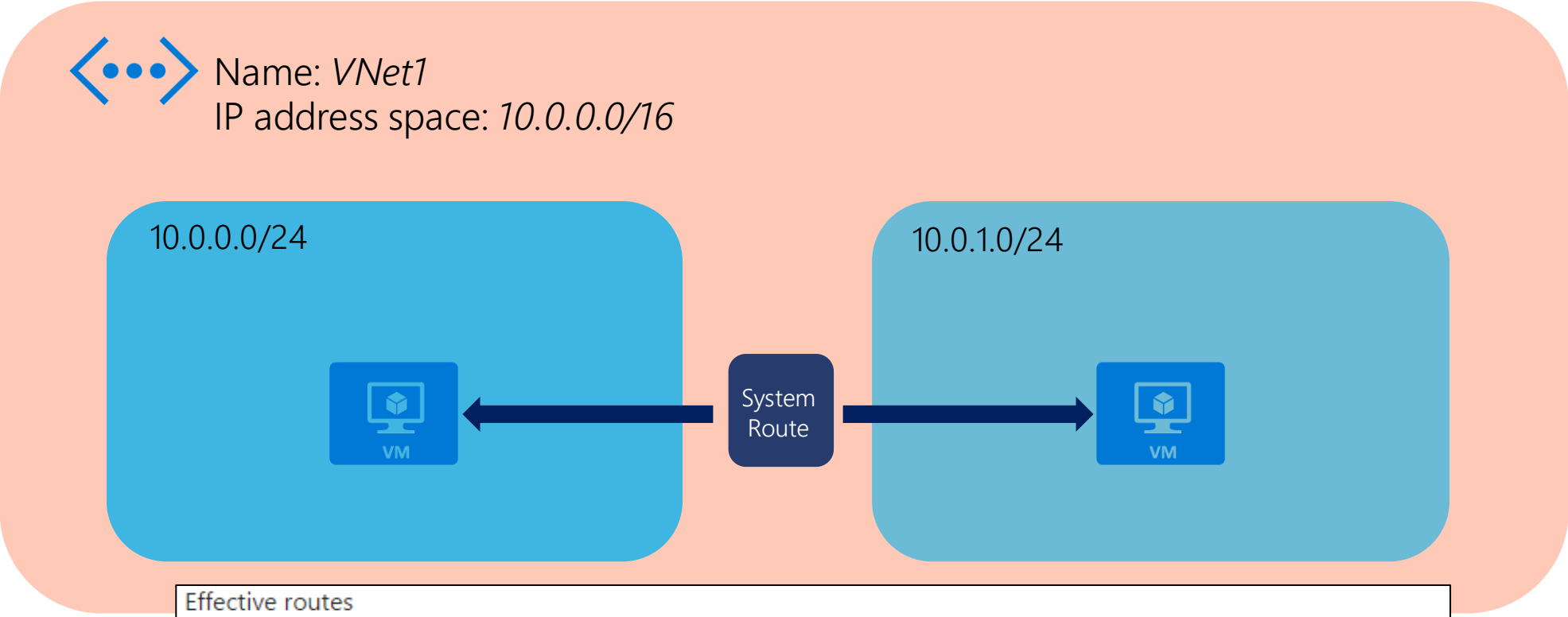


By default, VMs in different subnets within a VNet can route to each other directly.



A *system route* is installed to allow traffic within the Virtual Network

System routes are also installed for VNet peerings, VM to Internet connectivity, VPN gateway connectivity, etc.



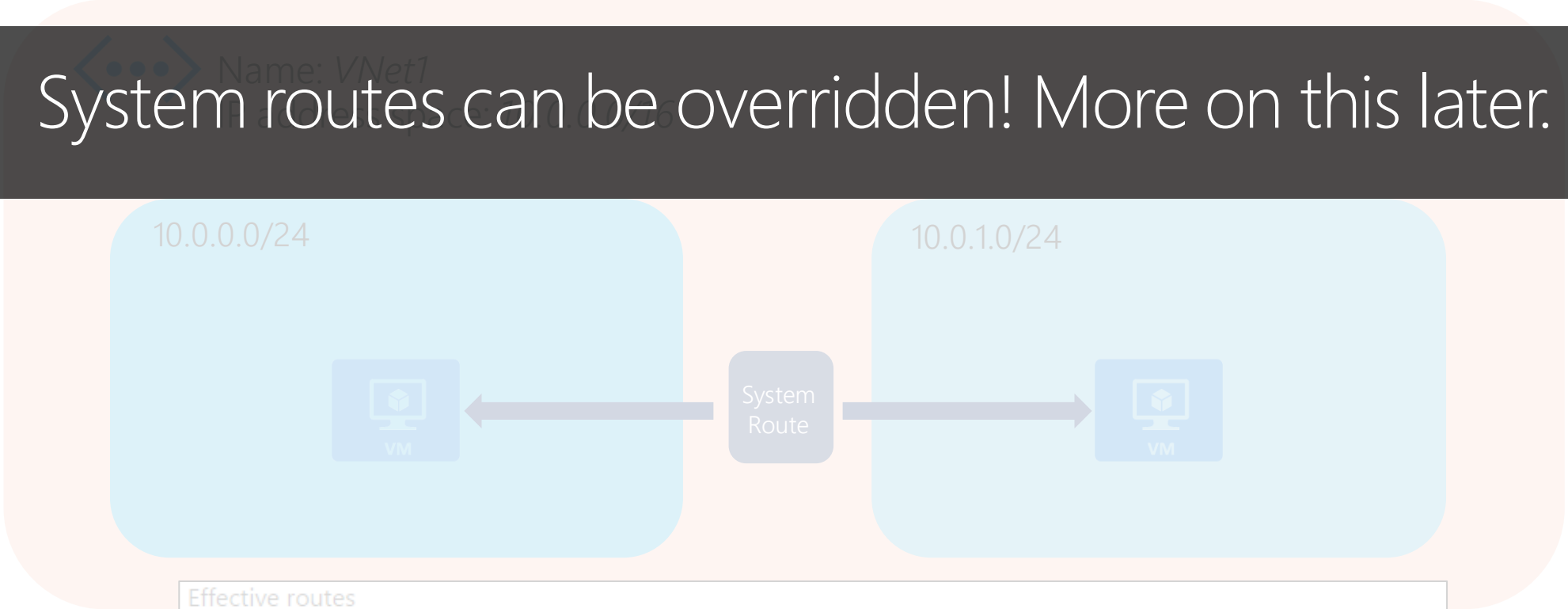
Effective routes

SOURCE	STATE	ADDRESS PREFIXES	NEXT HOP TYPE
Default	Active	10.0.0.0/16	Virtual network

A *system route* is installed to allow traffic within the Virtual Network

System routes are also installed for VNet peerings, VM to Internet connectivity, VPN gateway connectivity, etc.

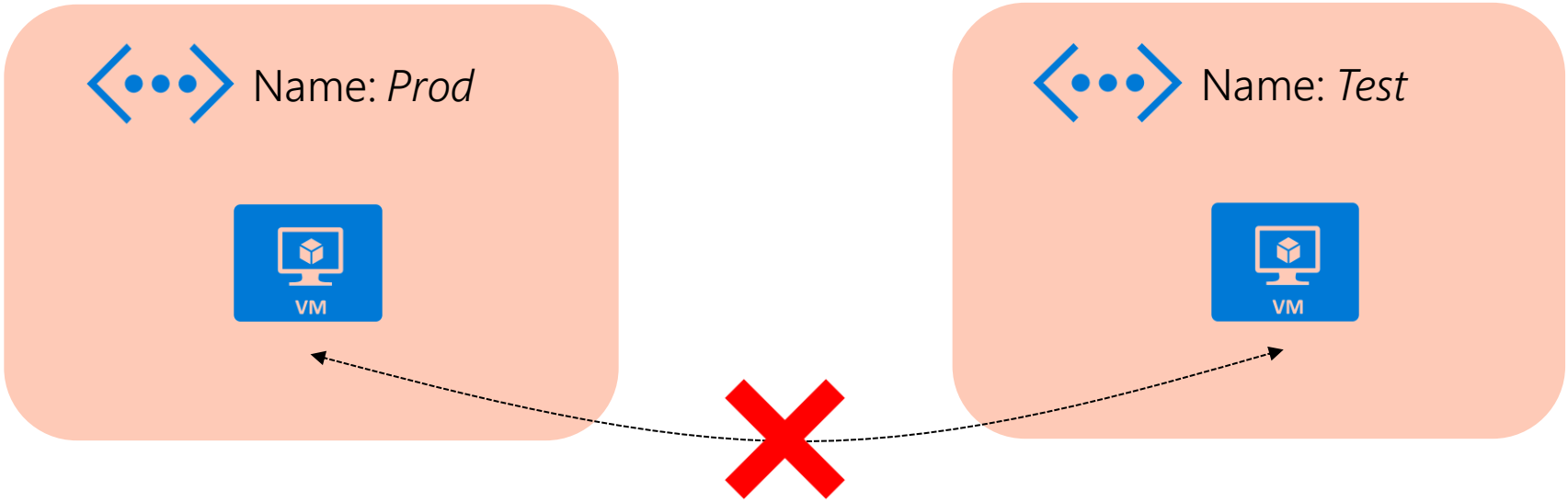
System routes can be overridden! More on this later.



Effective routes

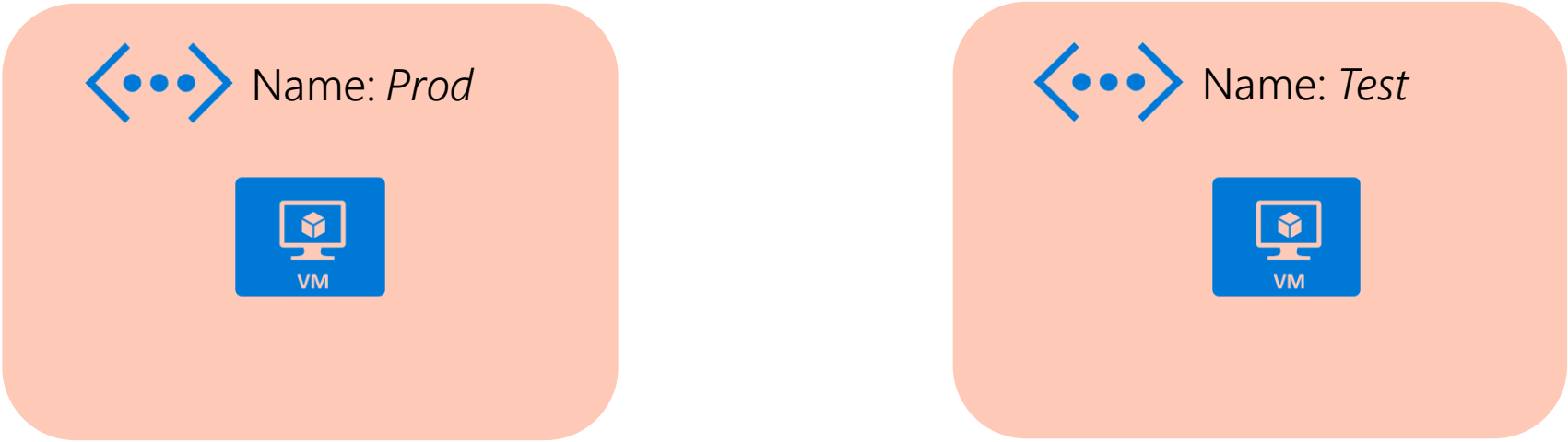
SOURCE	STATE	ADDRESS PREFIXES	NEXT HOP TYPE
Default	Active	10.0.0.0/16	Virtual network

VMs in different VNets *cannot* communicate, unless you configure this specifically (e.g. VNet peering or other method).



Connecting Virtual Networks Together

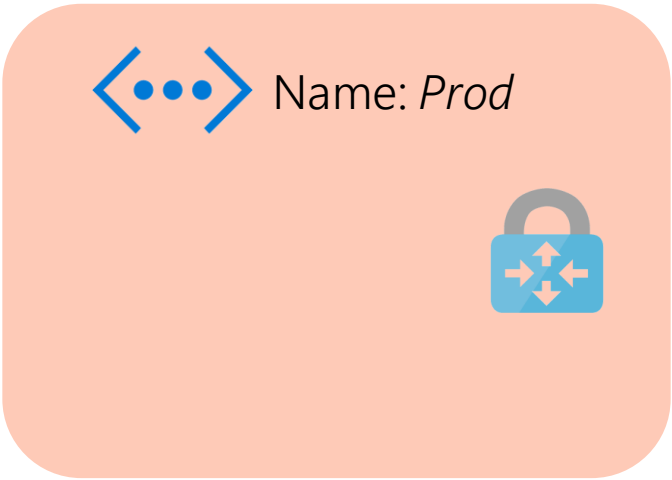
At some point, you'll want to allow connectivity between different virtual networks.



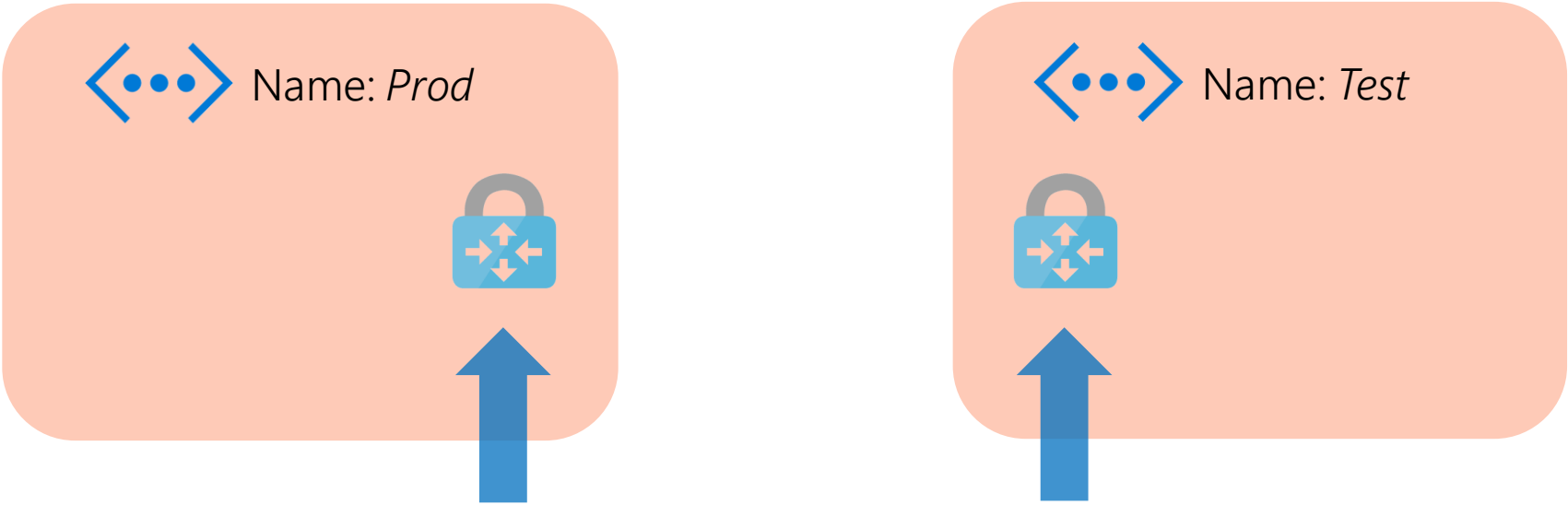
There are two main ways to achieve that:
VPN Gateways or Virtual Network Peering.

Option 1: VPN Gateways

VPN gateways are appliances that can be used to connect two VNets together, or between on-premises networks and Azure.

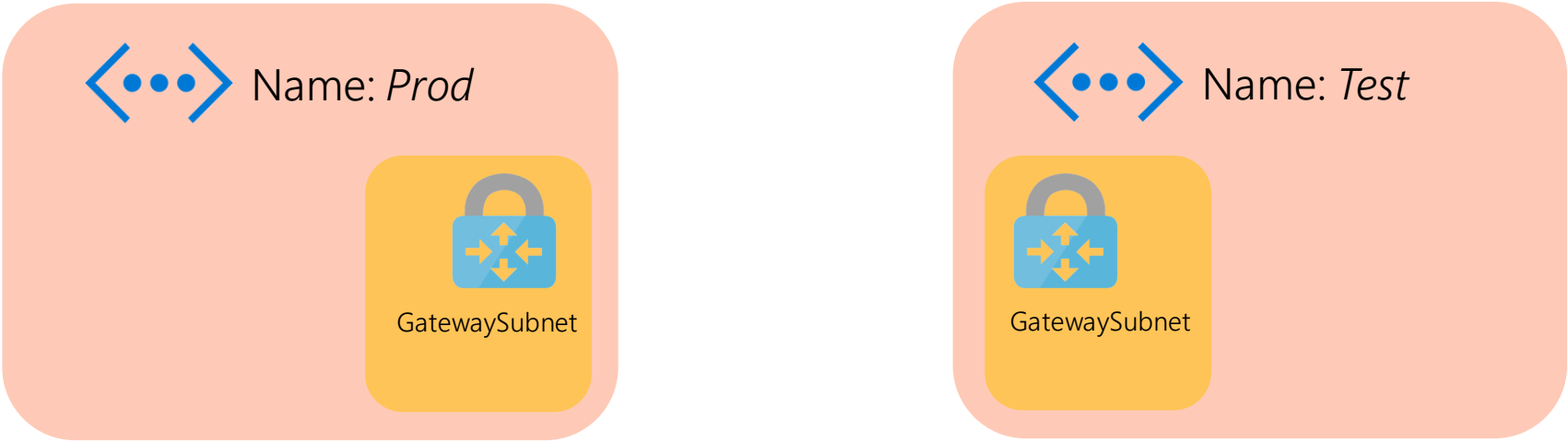


Option 1: VPN Gateways



To connect two VNets together,
you must create a VPN gateway in each VNet.

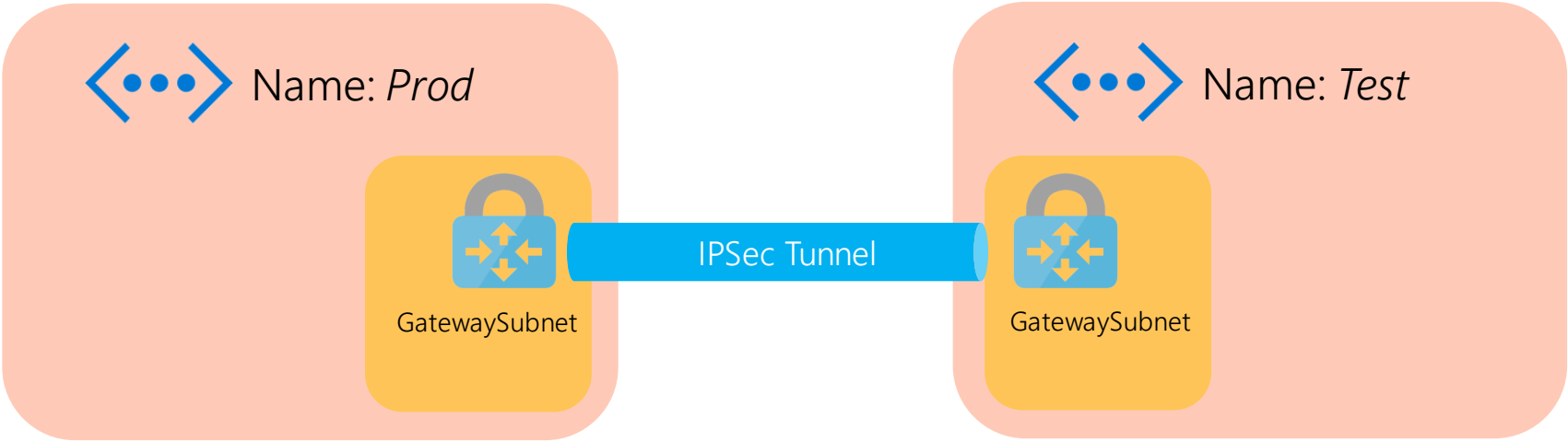
Option 1: VPN Gateways



VPN gateways always connect to a special subnet, called *GatewaySubnet* (this name is mandatory).

Option 1: VPN Gateways

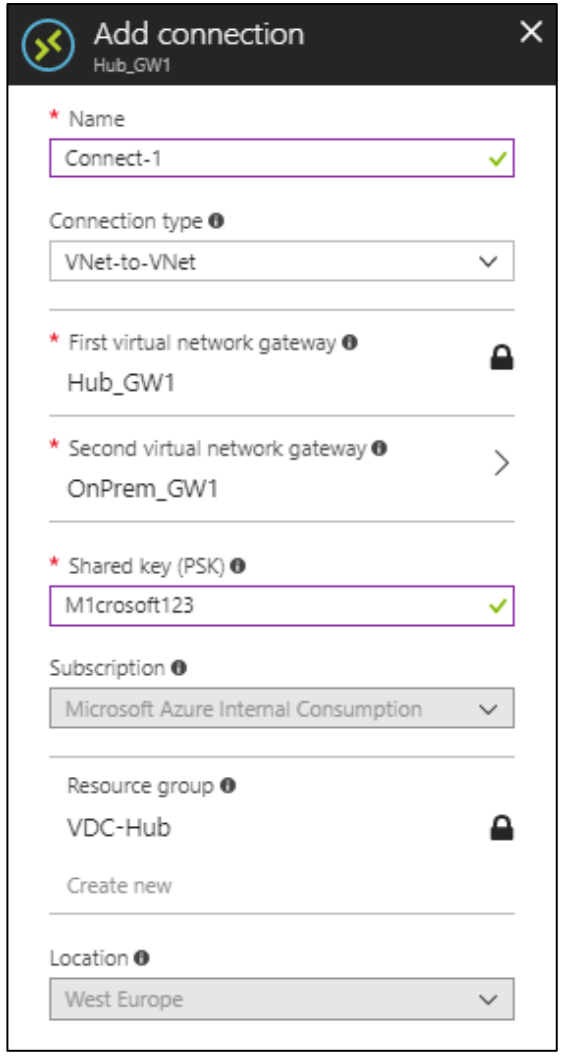
Once the gateways have been created,
the next step is to create a **connection** between them.



Option 1: VPN Gateways

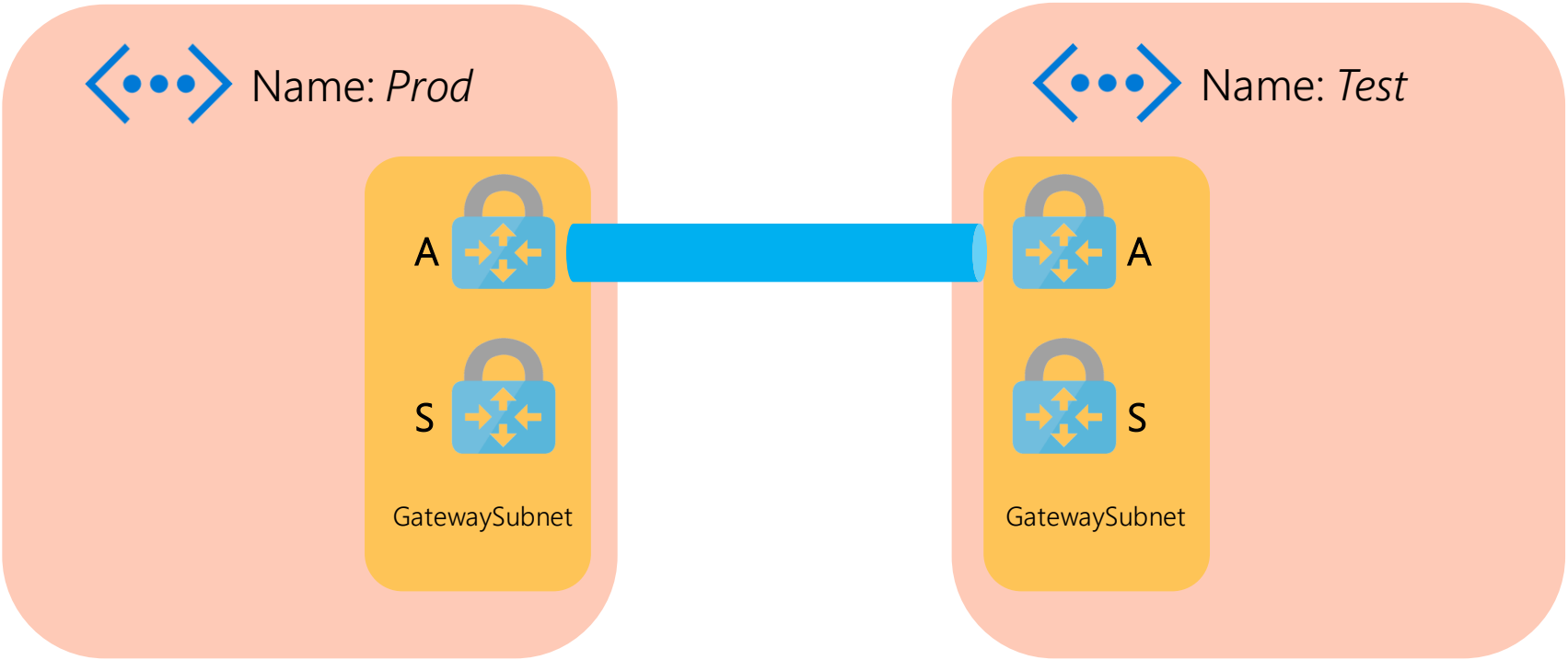
To create a connection, simply specify the two VPN gateways and configure a shared key.

You need to do this from both sides, i.e. create a connection from each VPN gateway to the other one.



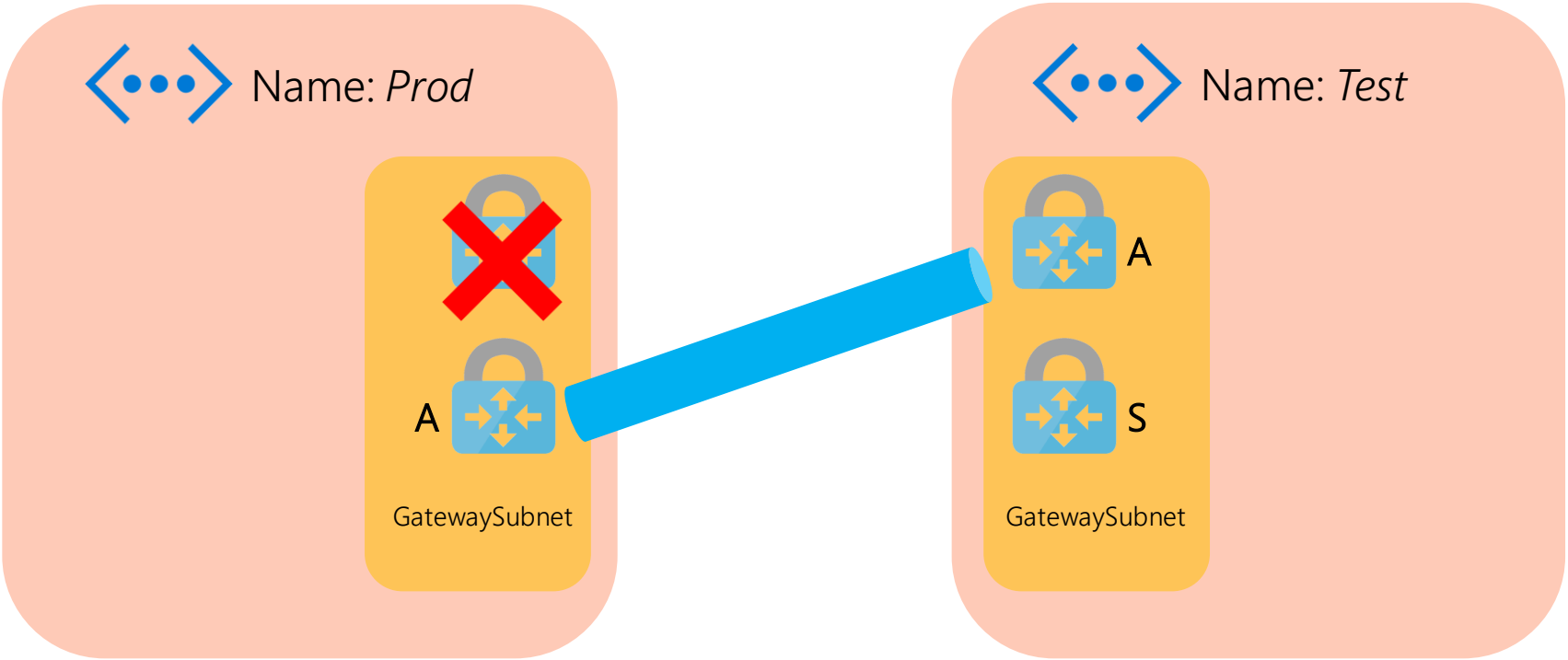
Option 1: VPN Gateways

Although this is hidden from the user, VPN gateways consist of two instances in an active / standby configuration.



Option 1: VPN Gateways

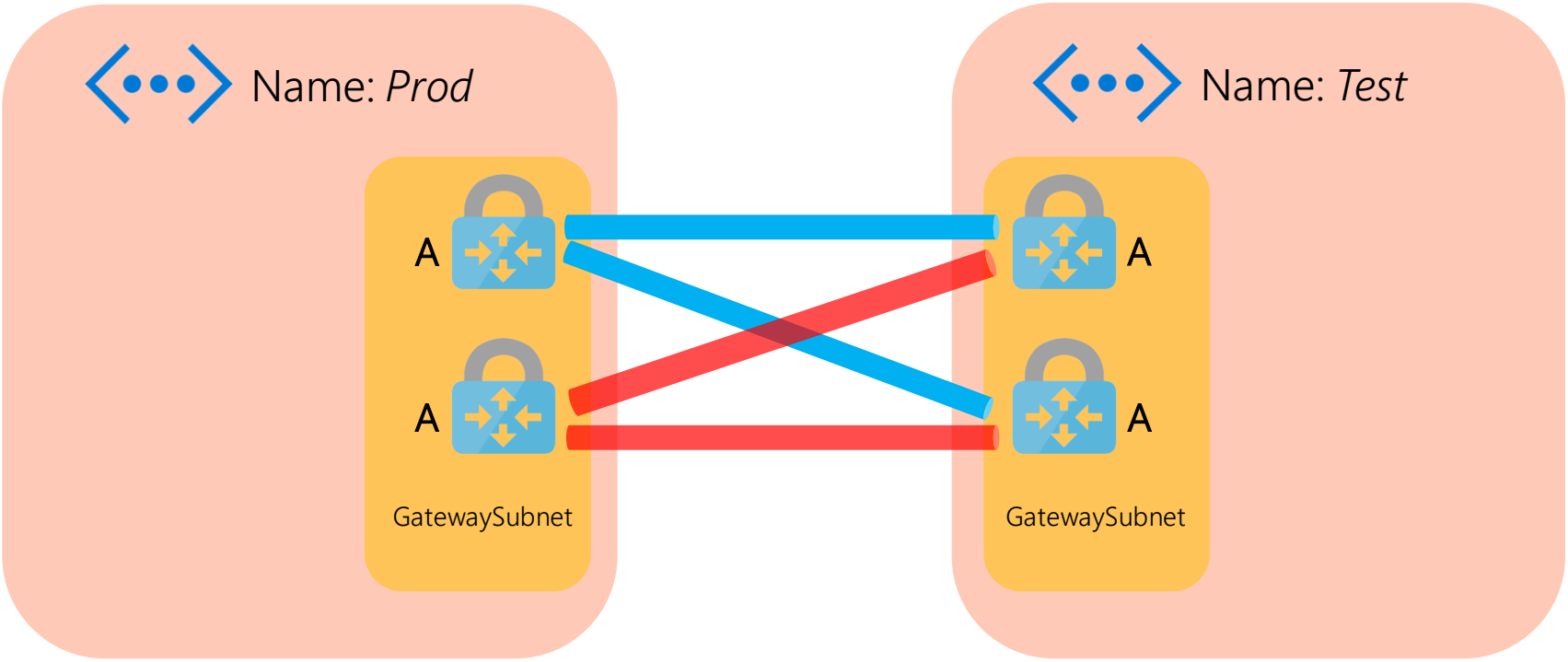
Failure of a gateway will result in the standby taking over (worst case 1.5 mins failover time).



Healthprobe URL: https://<ip>:8081/healthprobe

Option 1: VPN Gateways

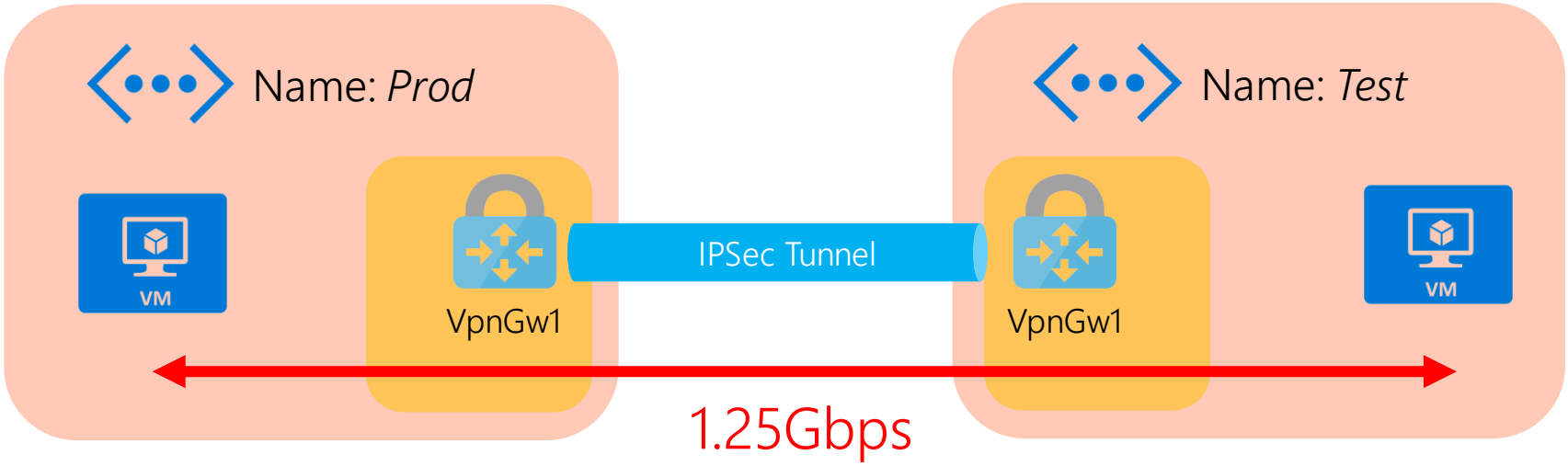
It's also possible to create VPN gateways in an active / active configuration, which will use a full mesh of IPSec tunnels.




```
New-AzureRmVirtualNetworkGateway -EnableActiveActiveFeature
```

Option 1: VPN Gateways

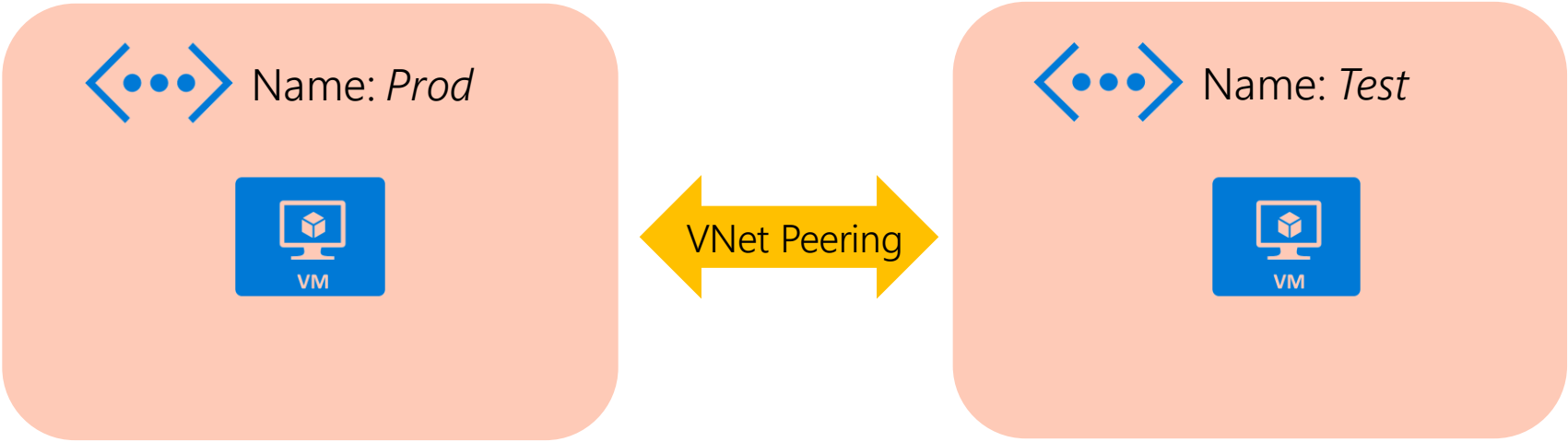
One of the issues with using VPN gateways to connect VNets is that you are limited to the bandwidth of the gateway (e.g. 650Mbps for VpnGw1).



VpnGw3

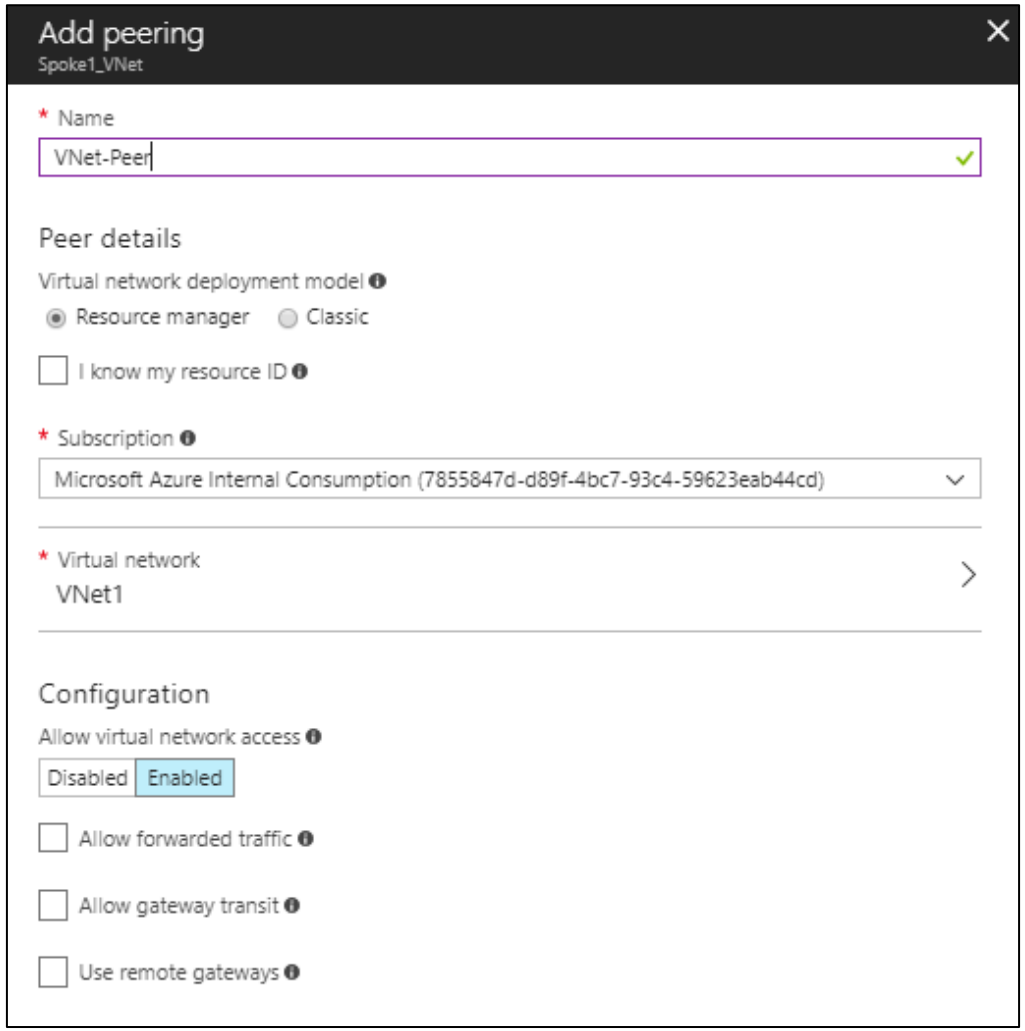
Option 2: VNet Peering

VNet Peering solved this limitation, it uses the Microsoft backbone network to connect the VNets together without the need of virtual gateways.

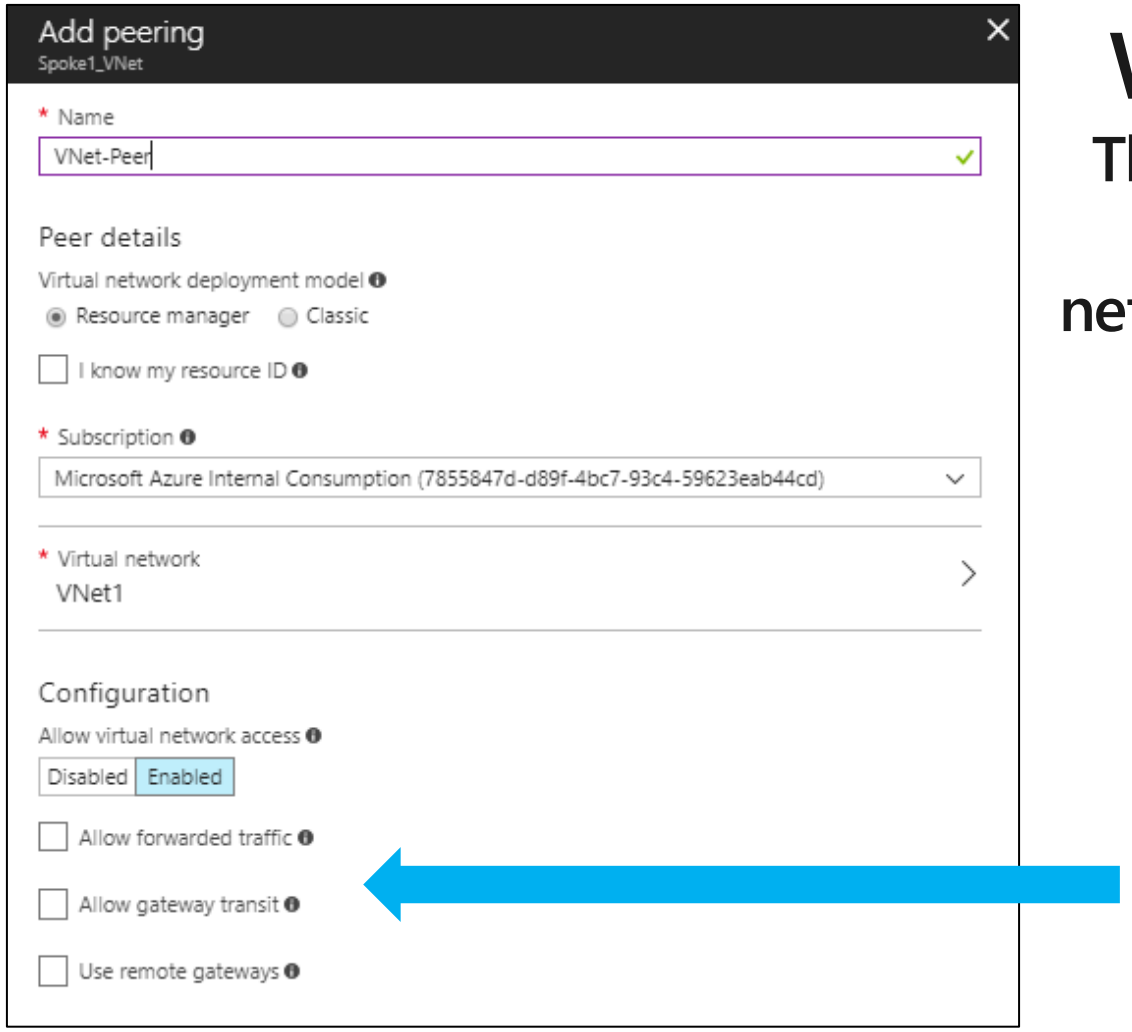


Option 2: VNet Peering

Setting up VNet Peering is easy – just specify the VNet you want to connect to.



Option 2: VNet Peering

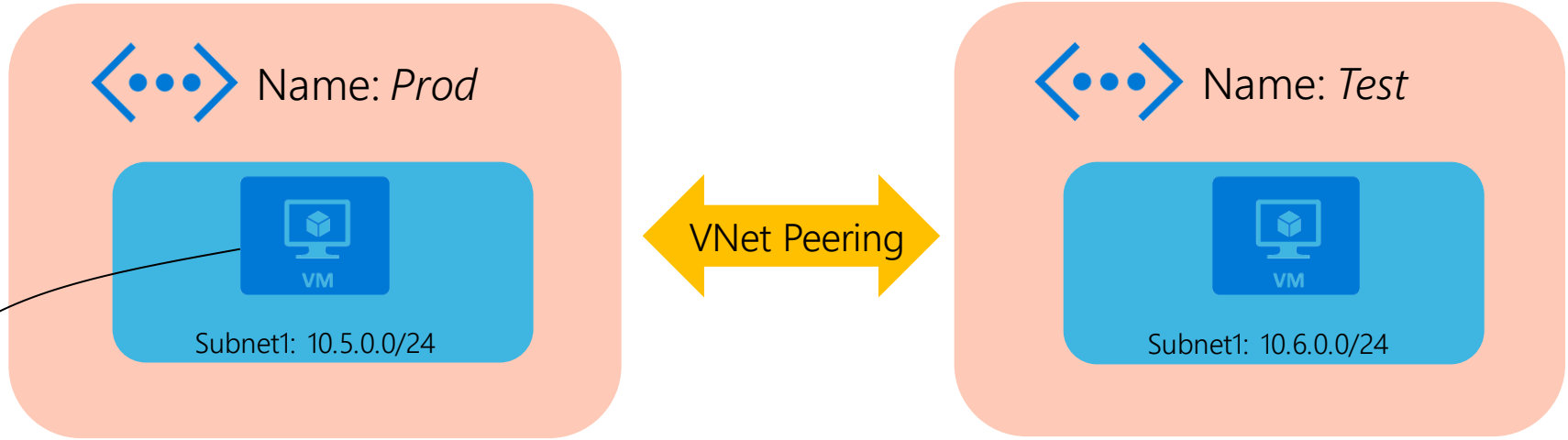


What are these options for?

These cover more advanced scenarios, such as sharing gateways, hub and spoke networks, etc – we'll come back to these later!

Option 2: VNet Peering

Once a VNet peering connection has been created, routes are automatically added to each VNet to point to the other.



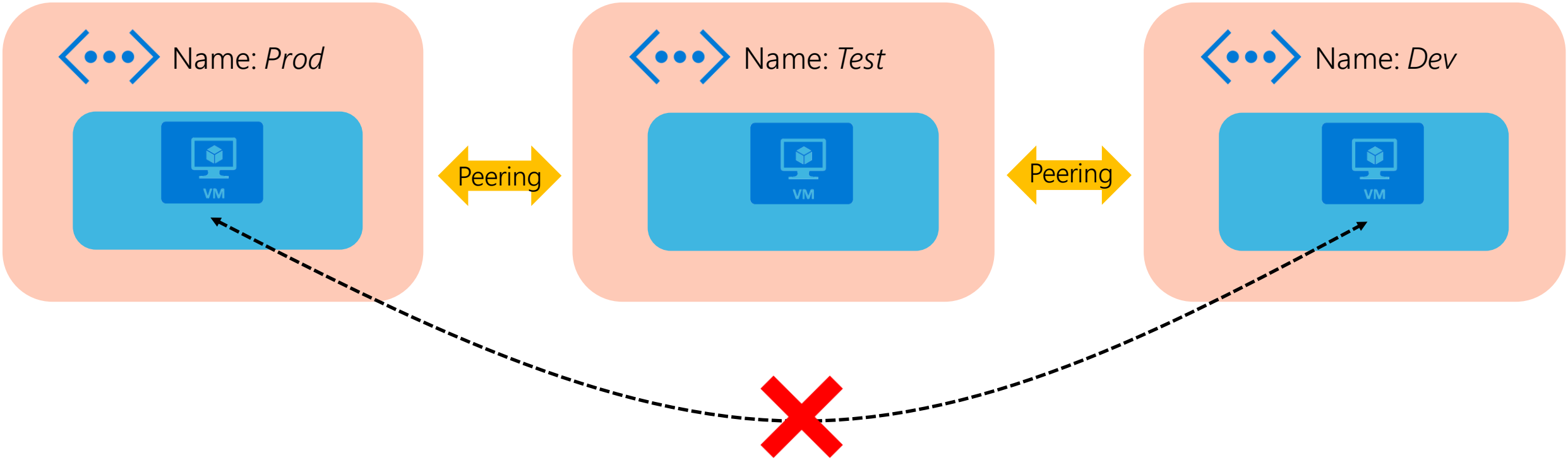
Effective routes

SOURCE	STATE	ADDRESS PREFIXES	NEXT HOP TYPE
Default	Active	10.5.0.0/24	Virtual network
Default	Active	10.6.0.0/24	VNet peering

You can find this by selecting 'Effective Routes' under the NIC connected to a VM inside the VNet.

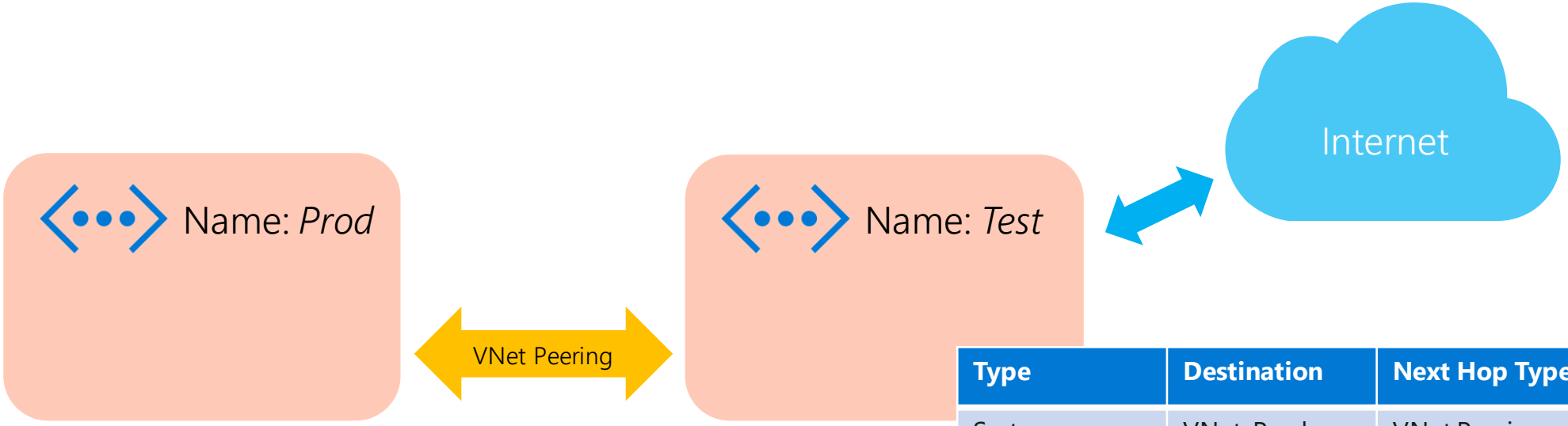
Option 2: VNet Peering

VNet Peerings are non-transitive – this means that VNet *Prod* cannot communicate with VNet *Dev* through VNet *Test*.



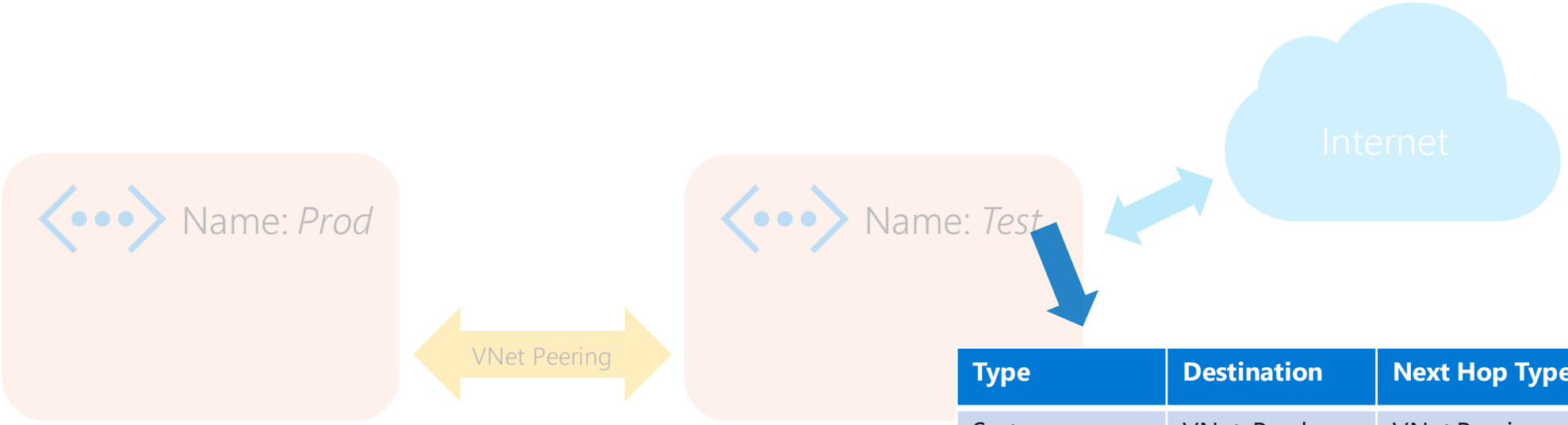
Routing in Azure

Azure configures System Routes when required – for example, for routing between subnets / VNets, routing to the Internet, etc.

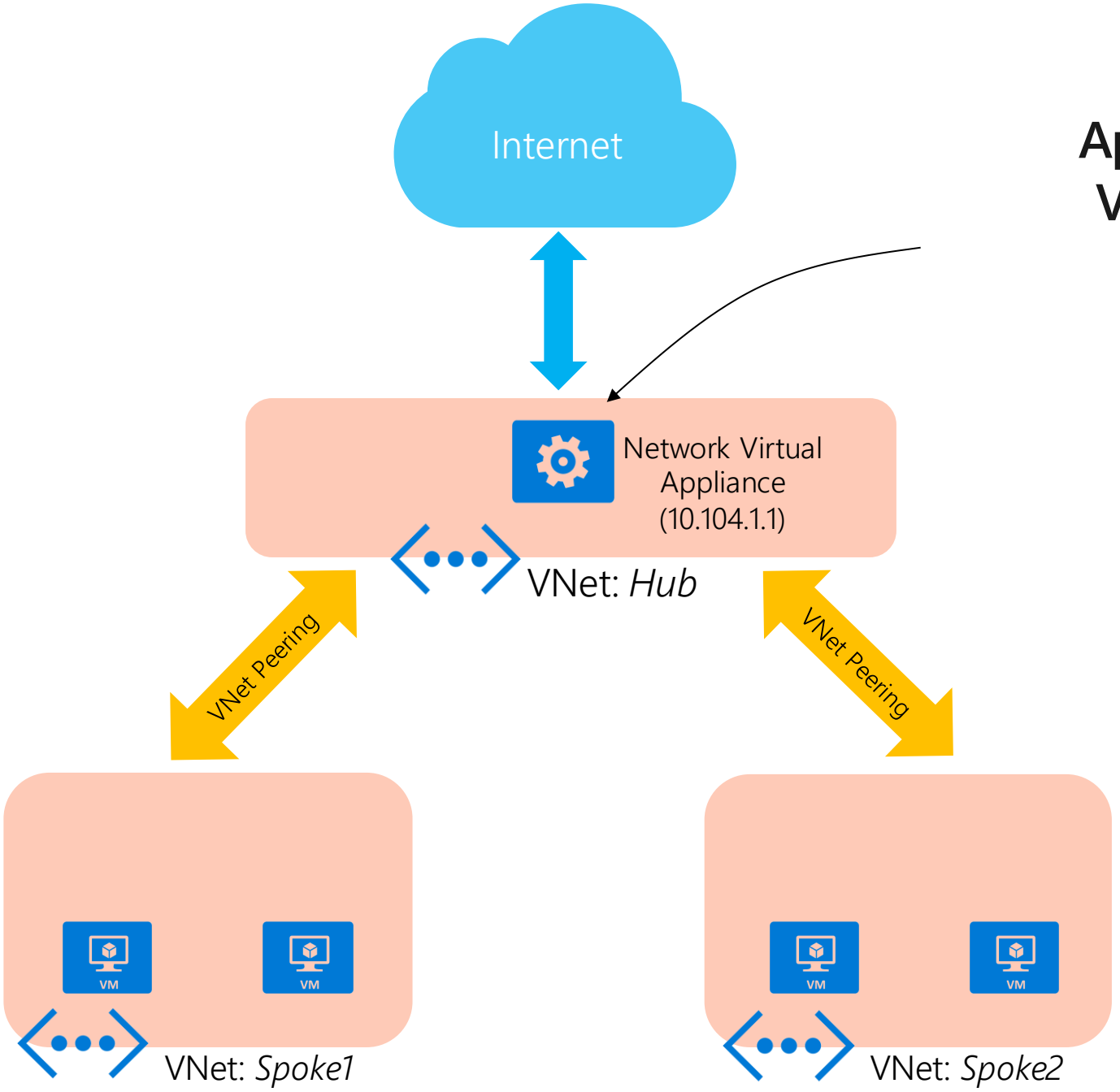


Type	Destination	Next Hop Type
System	VNet_Prod	VNet Peering
System	0.0.0.0	Internet

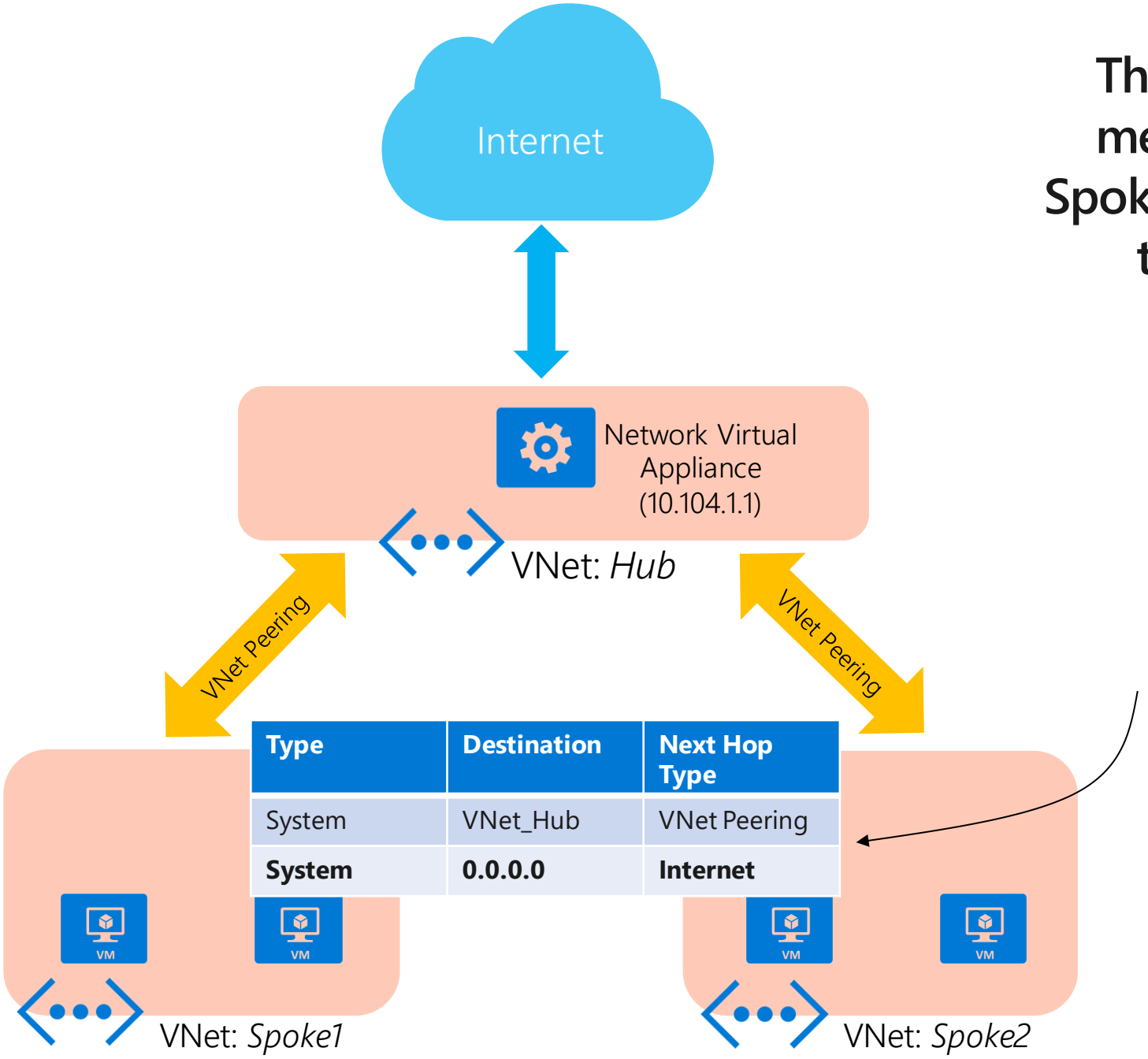
You can't delete System Routes, but you can *override* them, using User Defined Routes (UDRs).



Type	Destination	Next Hop Type
System	VNet_Prod	VNet Peering
System	0.0.0.0	Internet

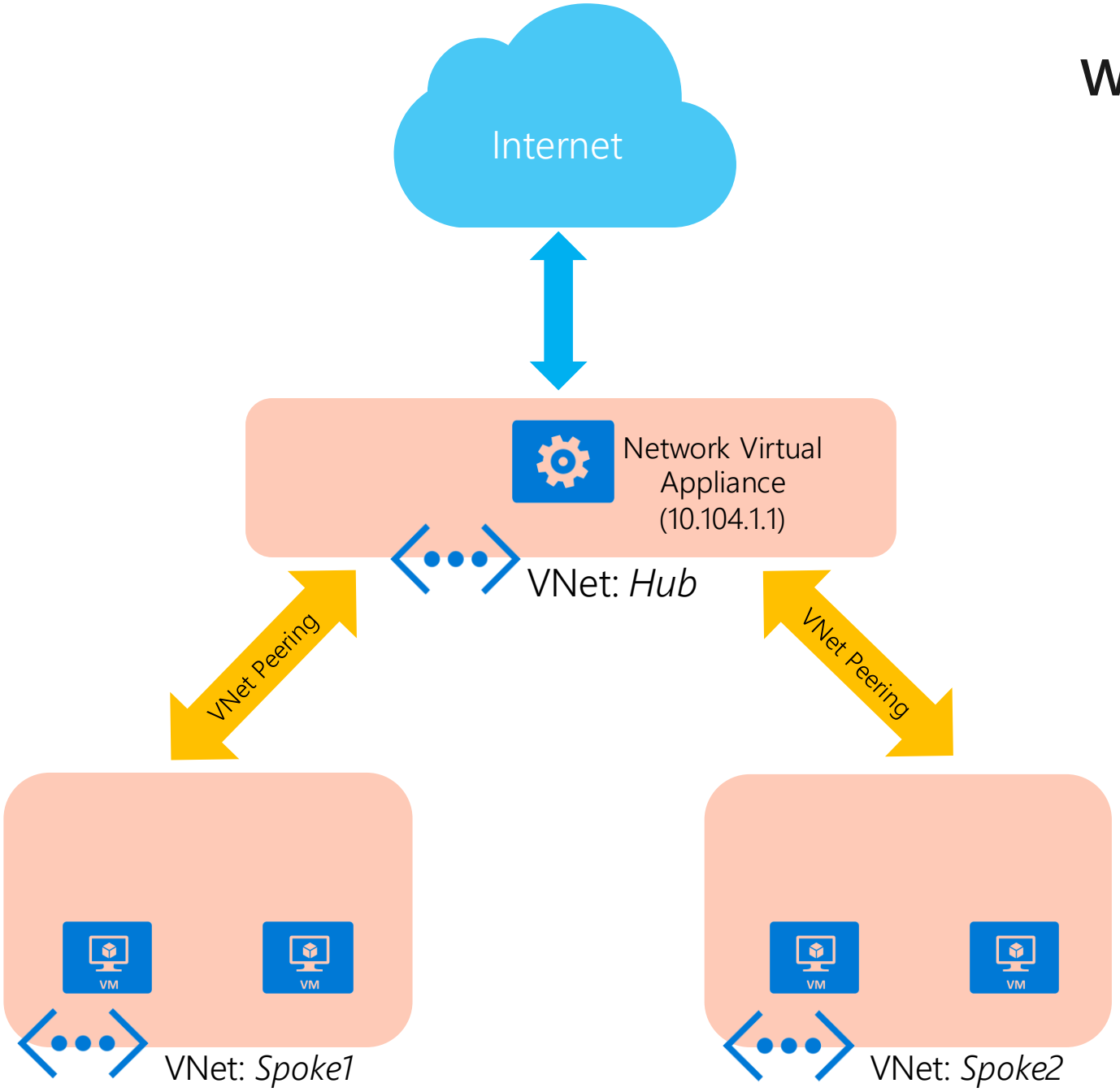


We have a Network Virtual Appliance (e.g. firewall) in the 'Hub' VNet, which we want to act as the enforcement point for Internet traffic.



Type	Destination	Next Hop Type
System	VNet_Hub	VNet Peering
System	**0.0.0.0**	**Internet**

The problem is, system routing means Internet traffic from the Spoke VNets goes directly out from the VNet (not via the NVA).



We configure a User Defined Route to point to the Internet via the virtual appliance.

Type	Destination	Next Hop Type	Next Hop
System	VNet_Hub	VNet Peering	
User Defined	**0.0.0.0**	**Virtual Appliance**	**10.104.1.1**

Then create routes

To configure UDRs, we first create a **Route Table**.

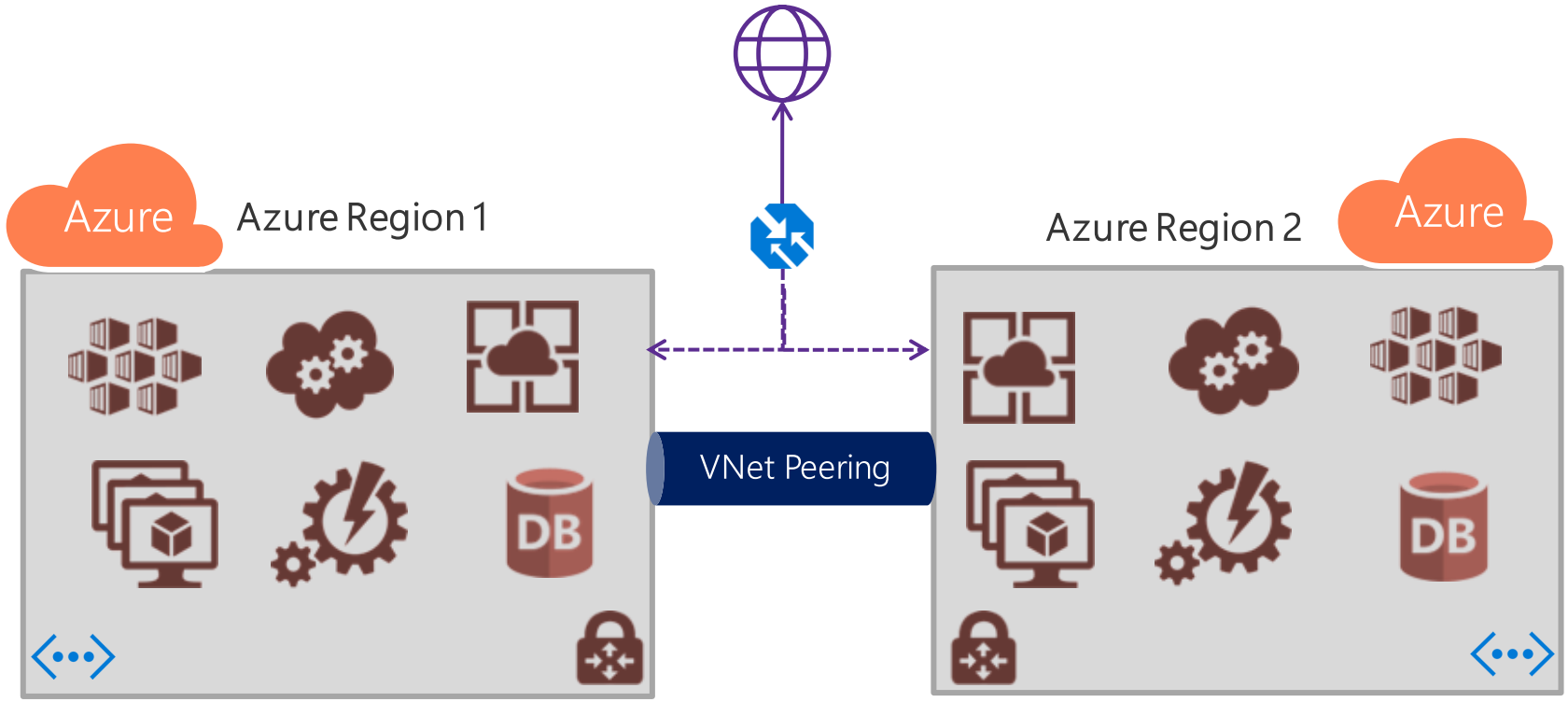
Type	Destination	Next Hop Type	Next Hop
User Defined	10.1.0.0/16	Virtual Appliance	10.104.1.1
User Defined	10.2.0.0/16	Virtual Appliance	10.104.1.1

Finally, we associate the route table with a subnet.

Type	Destination	Next Hop Type	Next Hop
User Defined	10.1.0.0/16	Virtual Appliance	10.104.1.1
User Defined	10.2.0.0/16	Virtual Appliance	10.104.1.1

Subnet1: 10.0.0.0/24

Global VNet Peering

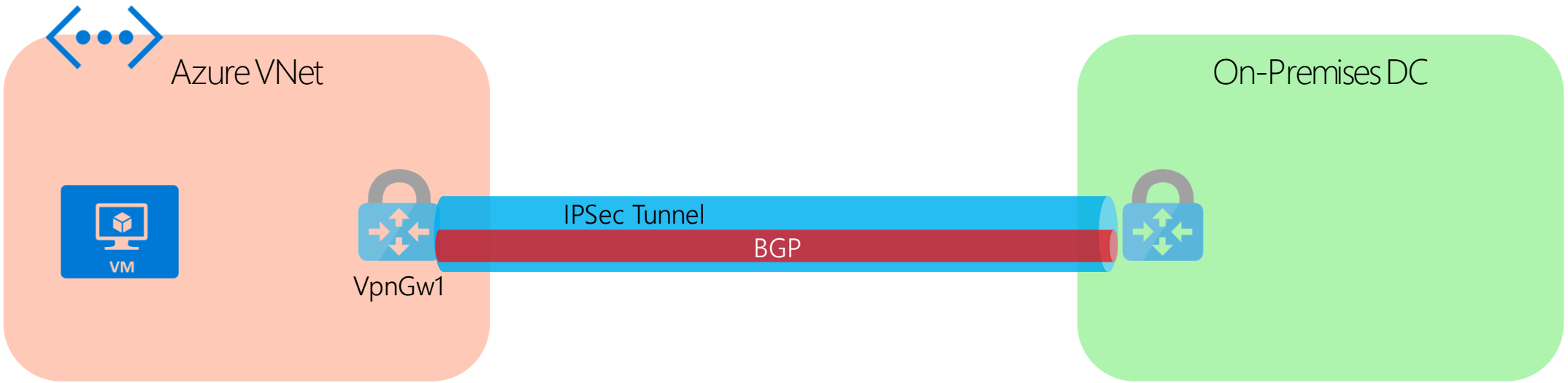


VNet Peering

- Vnets in different regions connected over Azure backbone network
- Simple to setup, configure in minutes
- Enables cross region data center replication over private IP address
- Generally Available in all regions
- You can globally peer across subscriptions
- You can't have overlapping address spaces.
- No bandwidth limitations
- No extra hops
- **Now available in Azure China regions**

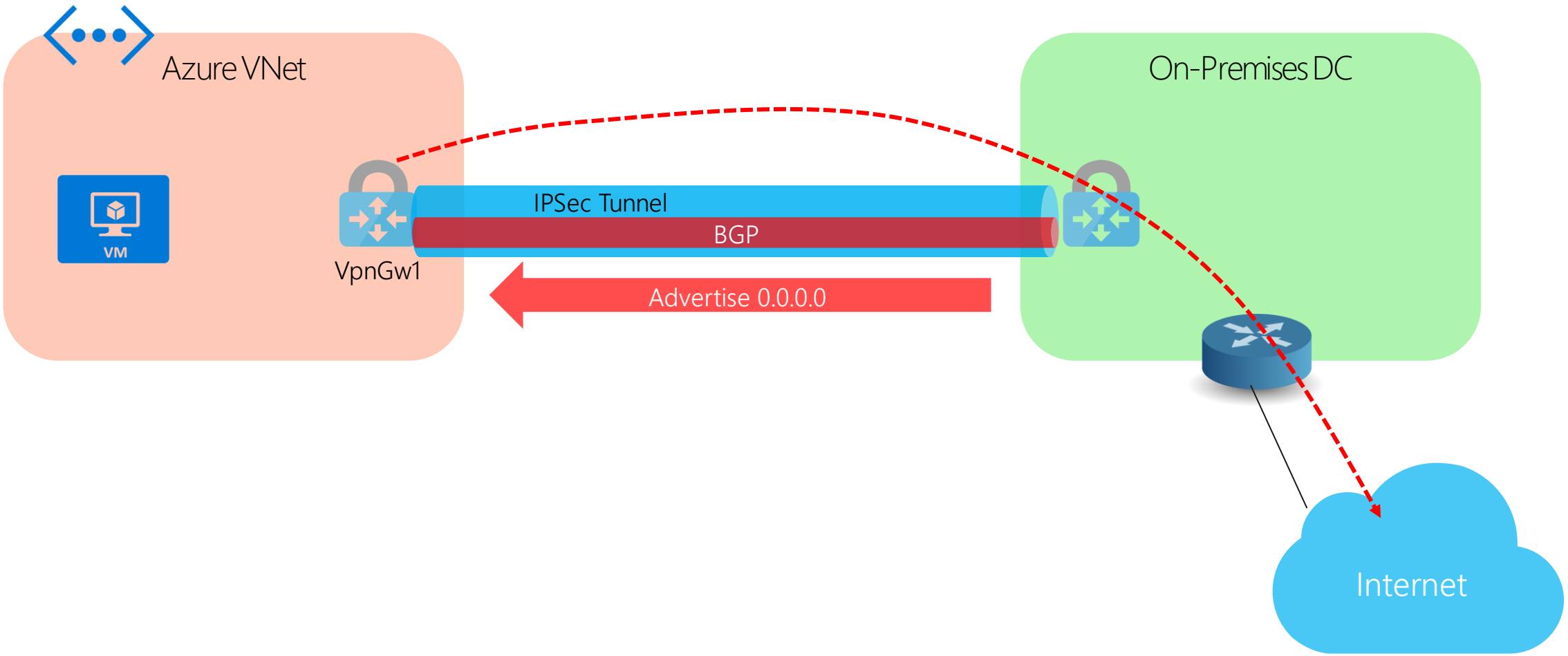
What about dynamic routing?
Is that supported?

BGP is supported on VPN Gateways*.



* Not on 'Basic' SKU or on 'policy-based' gateways

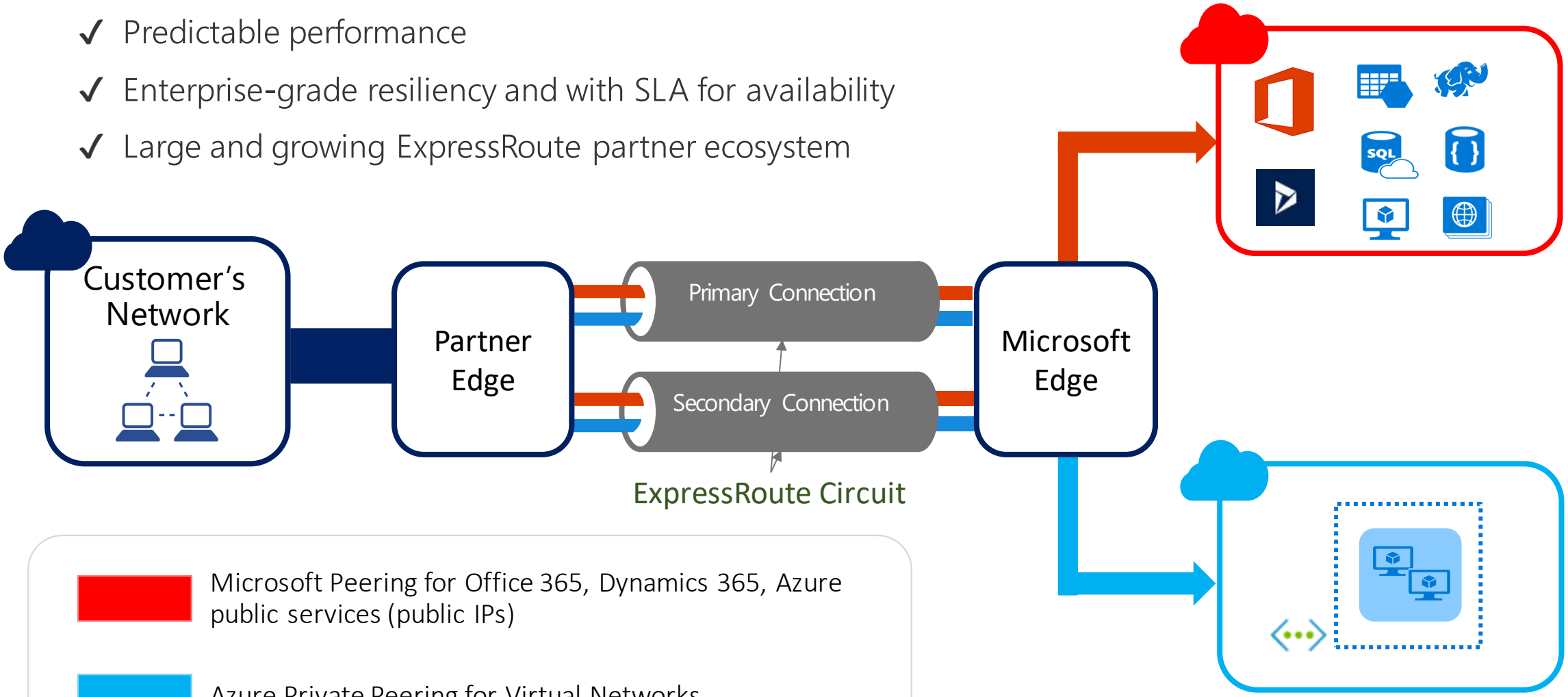
Advertising 0.0.0.0 to Azure forces all traffic via the VPN gateway.



Connecting with Express Route

ExpressRoute

- ✓ Unified connectivity to Microsoft Cloud Services
- ✓ Predictable performance
- ✓ Enterprise-grade resiliency and with SLA for availability
- ✓ Large and growing ExpressRoute partner ecosystem

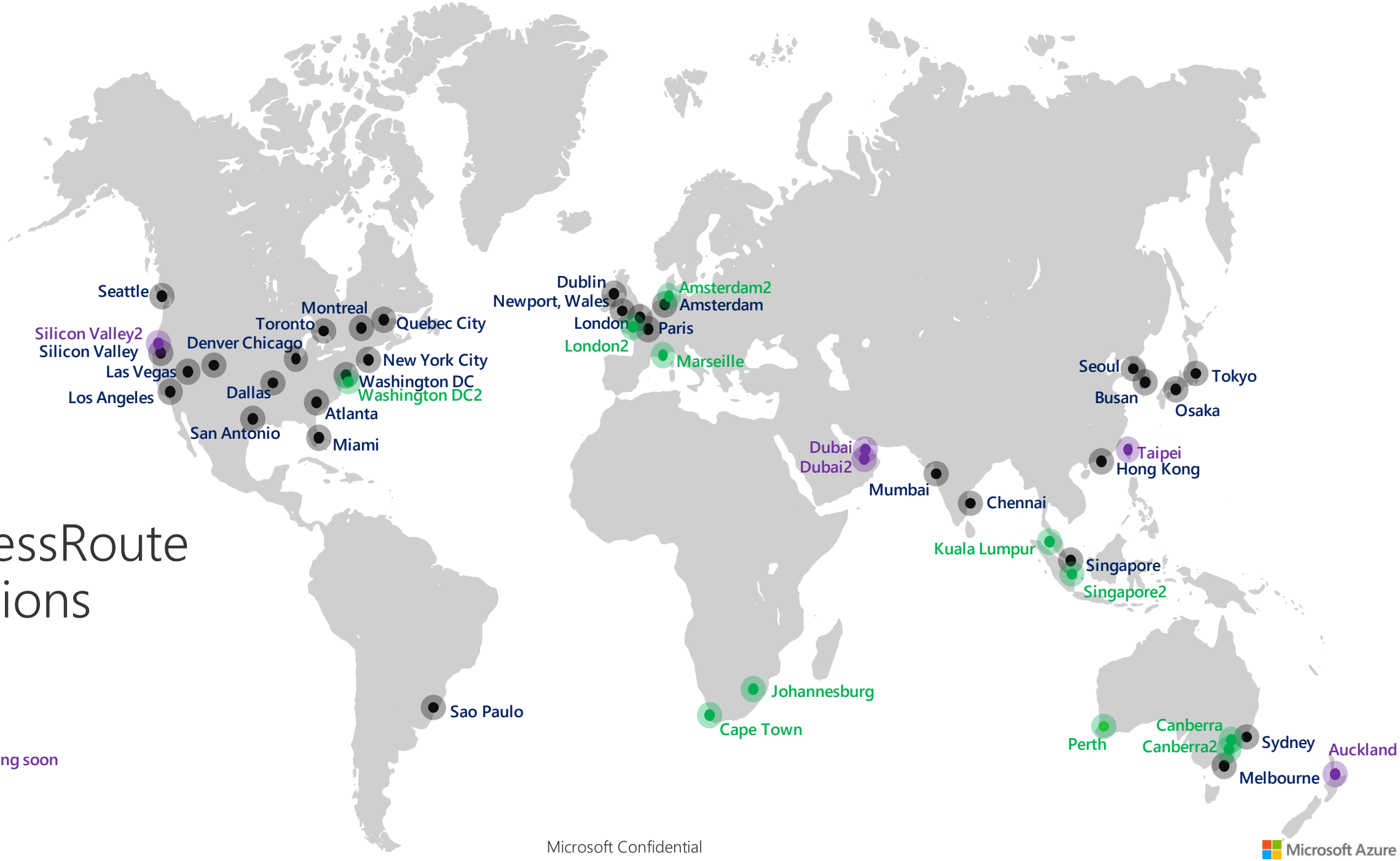


Microsoft Peering for Office 365, Dynamics 365, Azure public services (public IPs)

Azure Private Peering for Virtual Networks

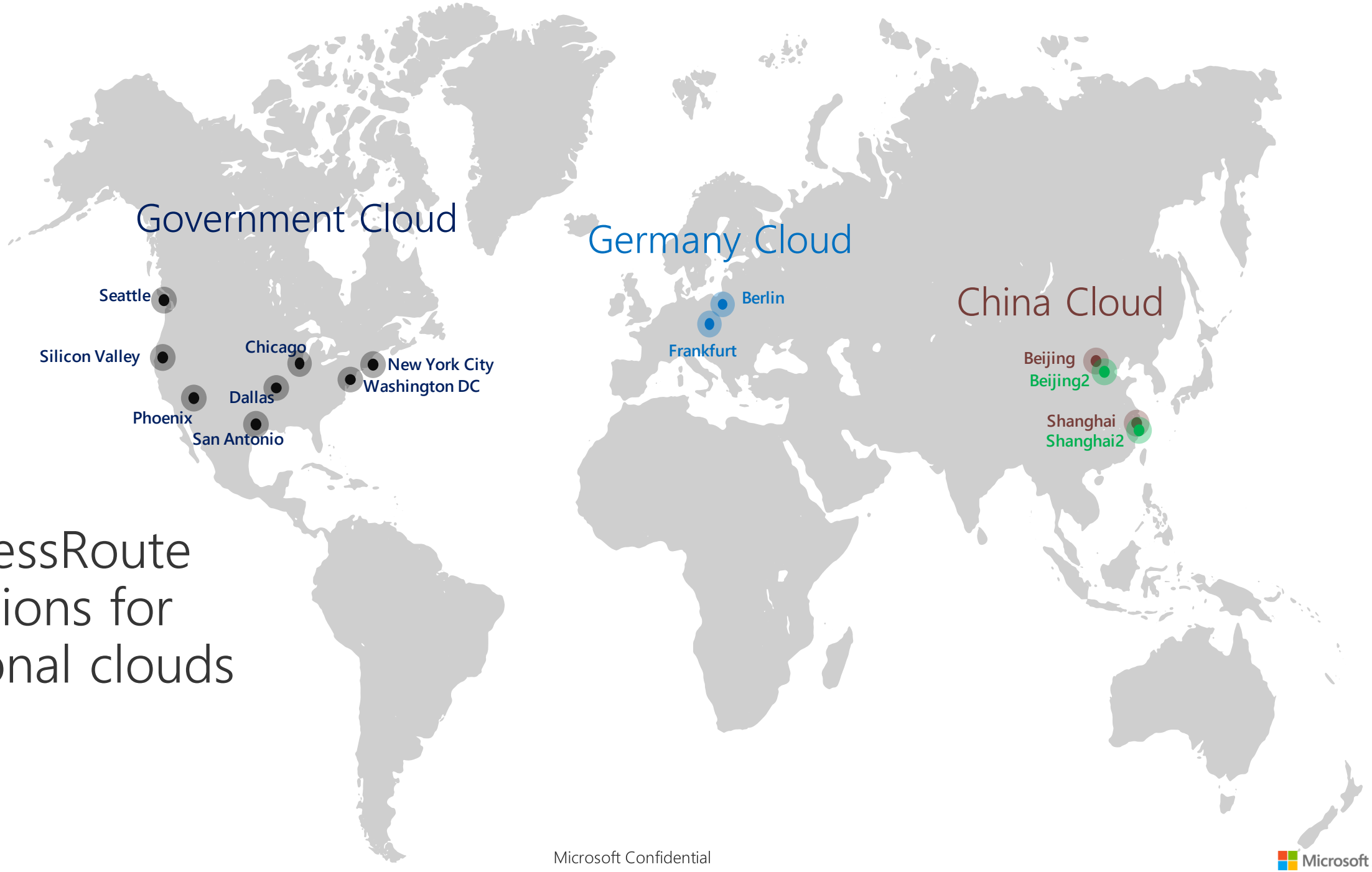
Microsoft Azure

ExpressRoute locations



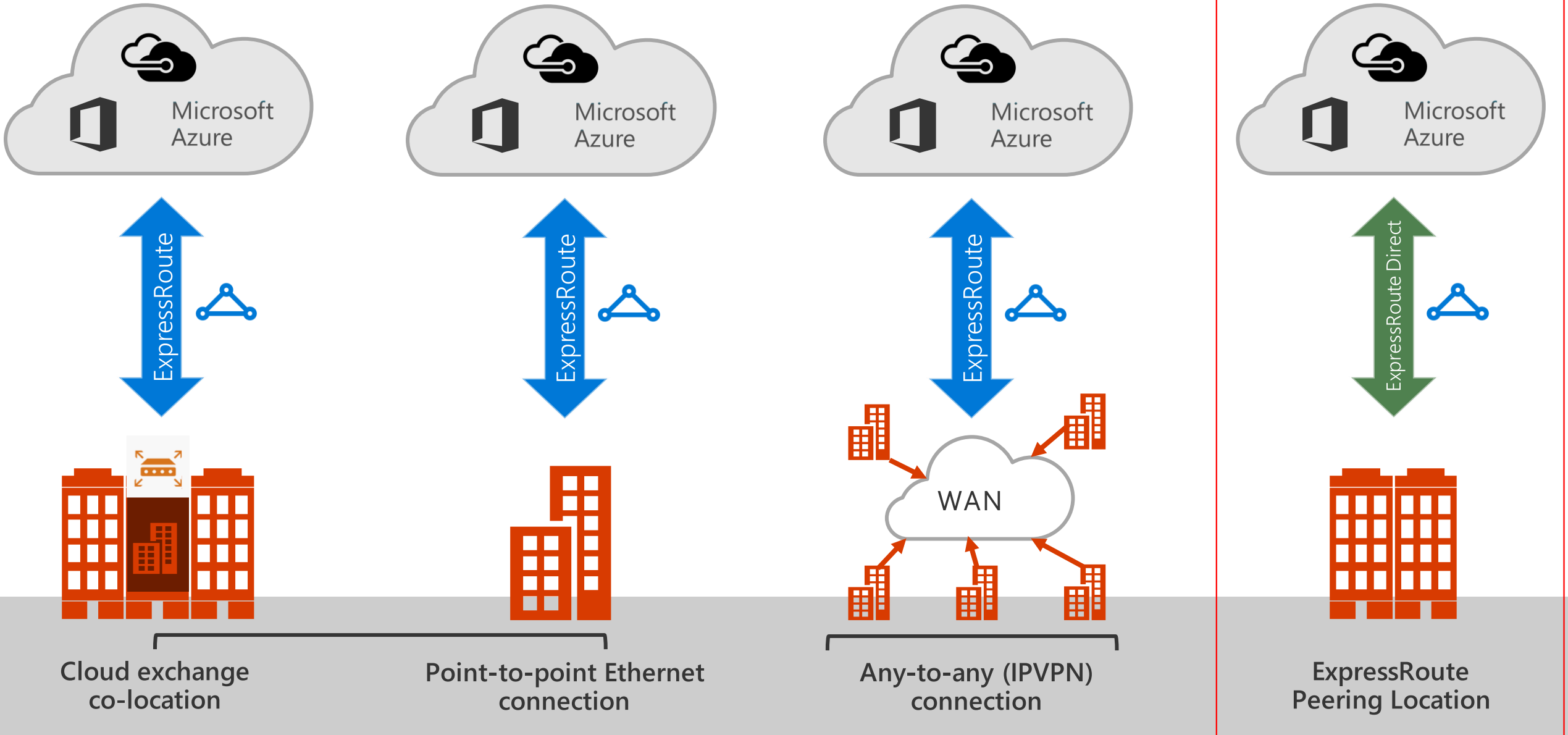
Microsoft Azure

ExpressRoute locations for national clouds



Microsoft Azure

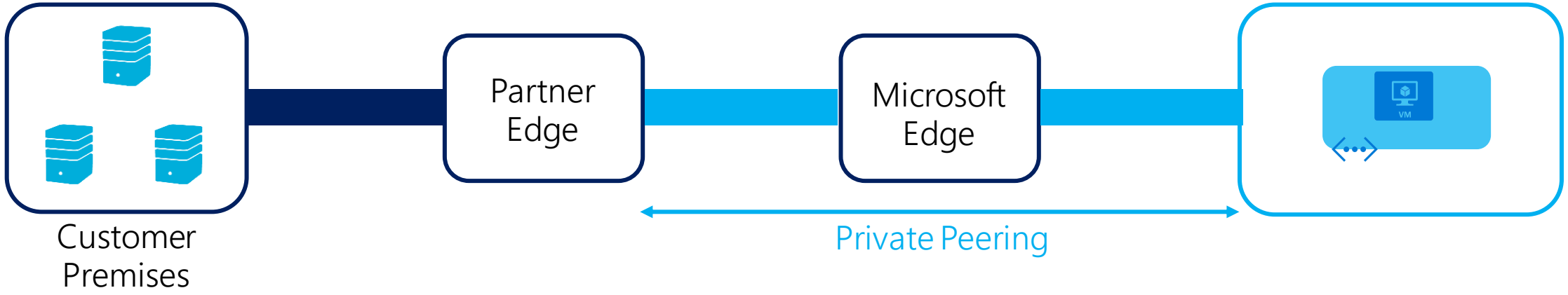
Express Route Connectivity Models



Two Peering Types

1. Private Peering

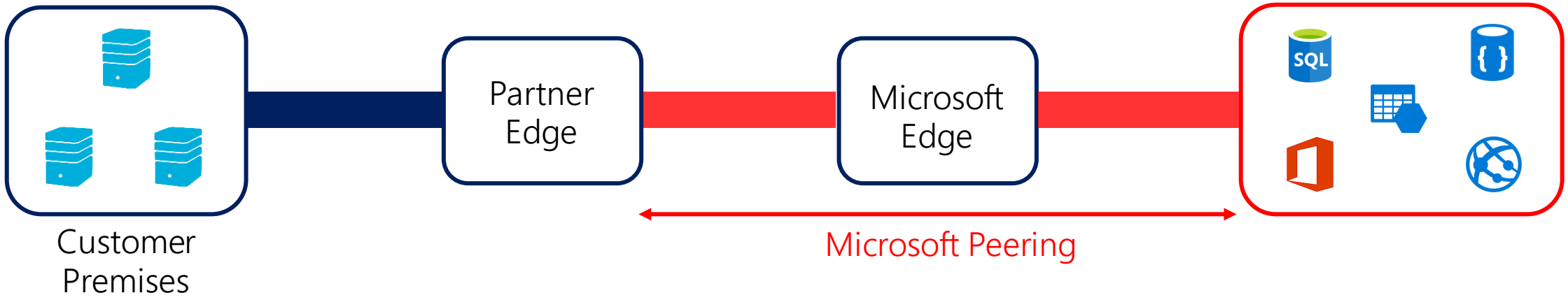
Connect to Azure services deployed within a virtual network.



Two Peering Types

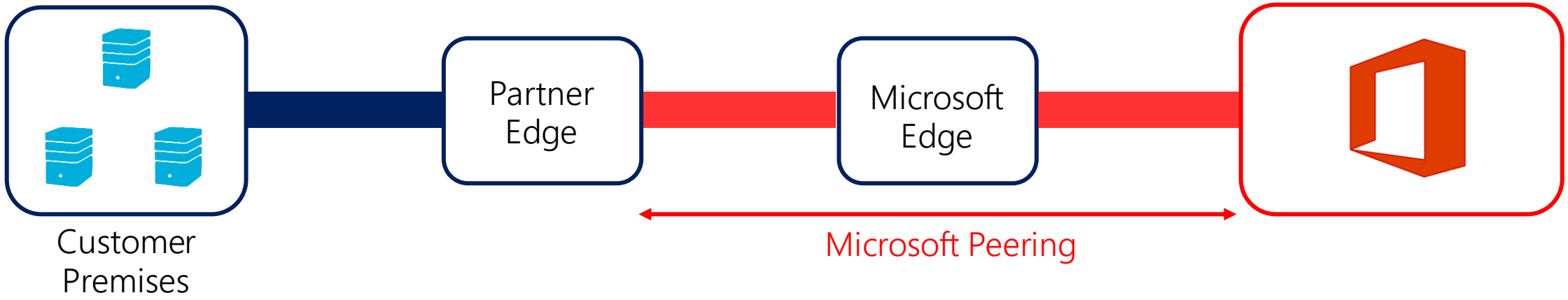
2. Microsoft Peering

Connect to Azure PaaS services (storage, SQL, Web Apps, etc) and SaaS services (O365, etc).

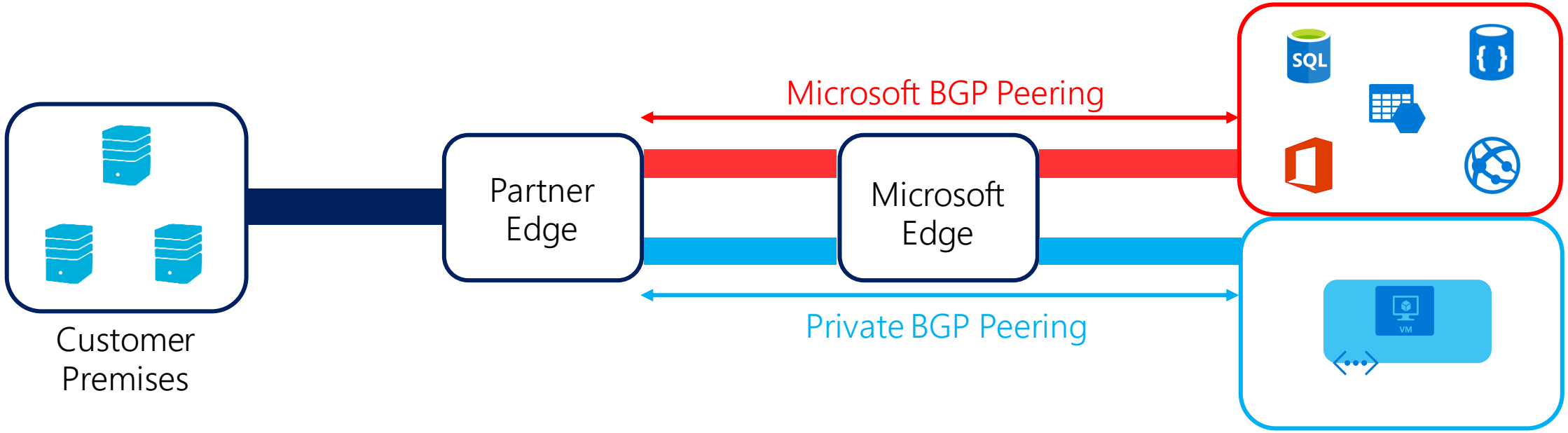


Two Peering Types

Important! Connecting to Office 365 via ExpressRoute Microsoft peering is NOT recommended and requires approval.



Each peering type requires a separate set of BGP peer sessions.



ExpressRoute Direct

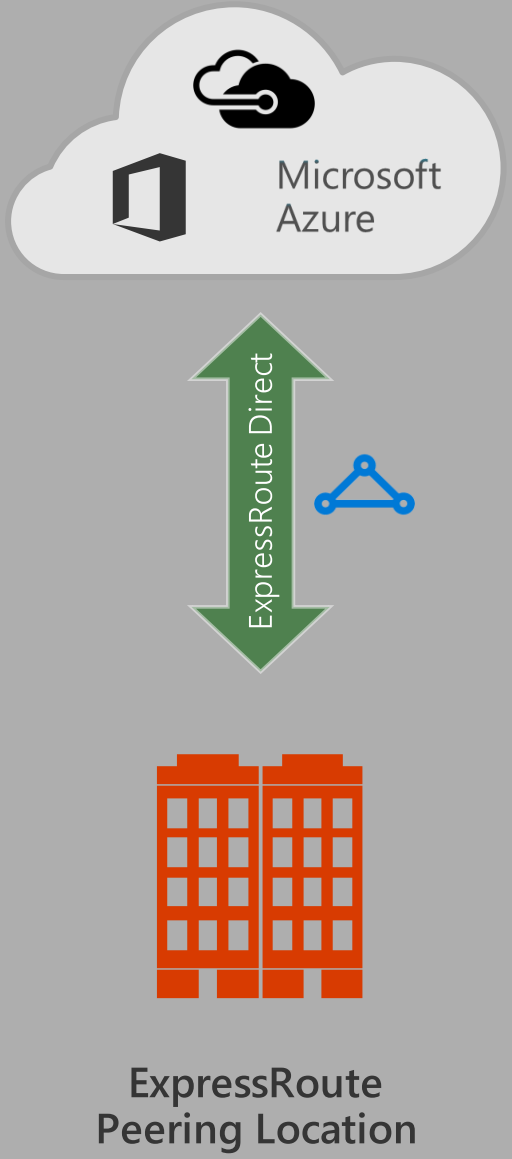
Fastest private connectivity in public cloud – 100Gbps to Azure

Customers given LOA for dual 100G connections to ExpressRoute routers

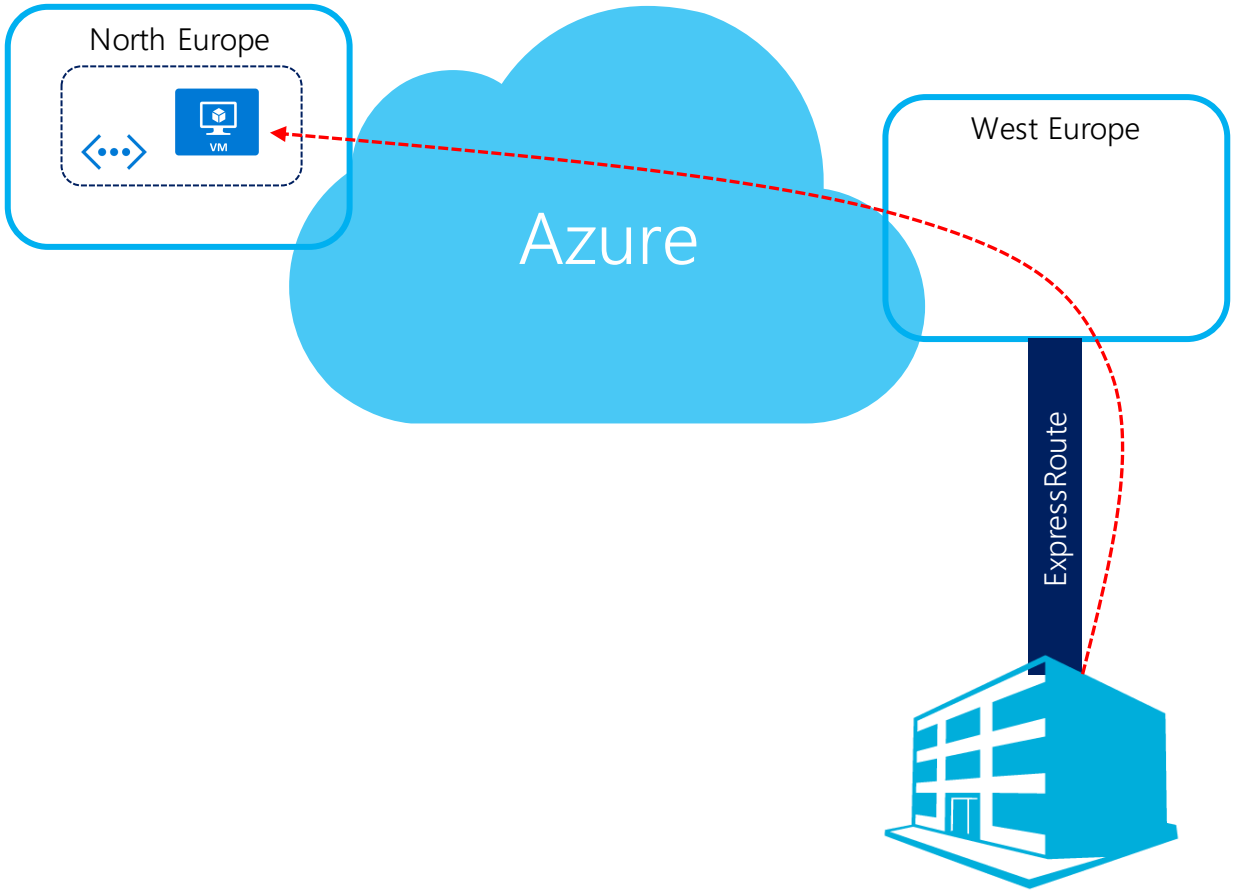
Customer demand due to compliance

Optimized for massive data ingestion to Azure Storage, Cosmos DB, Azure SQL

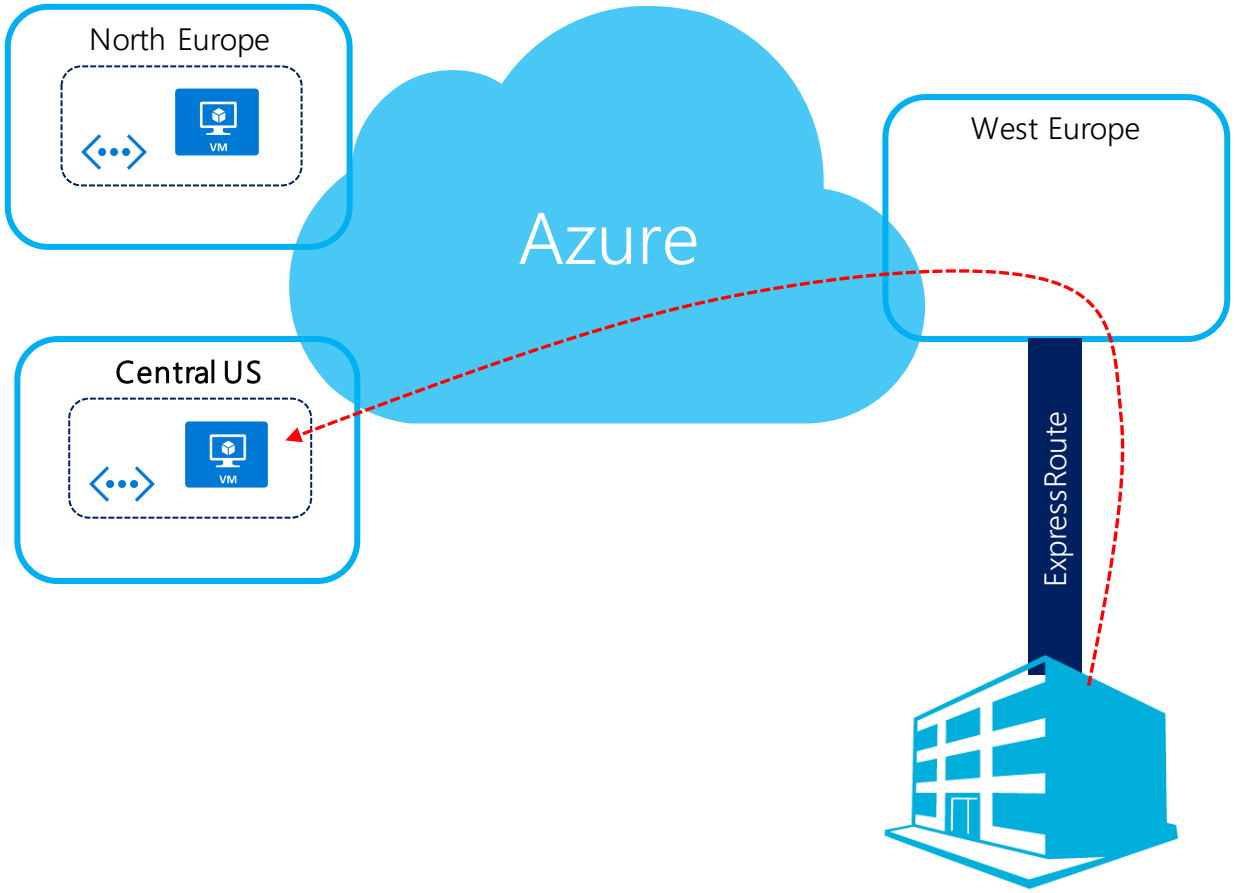
New 40G, 100G virtual circuits



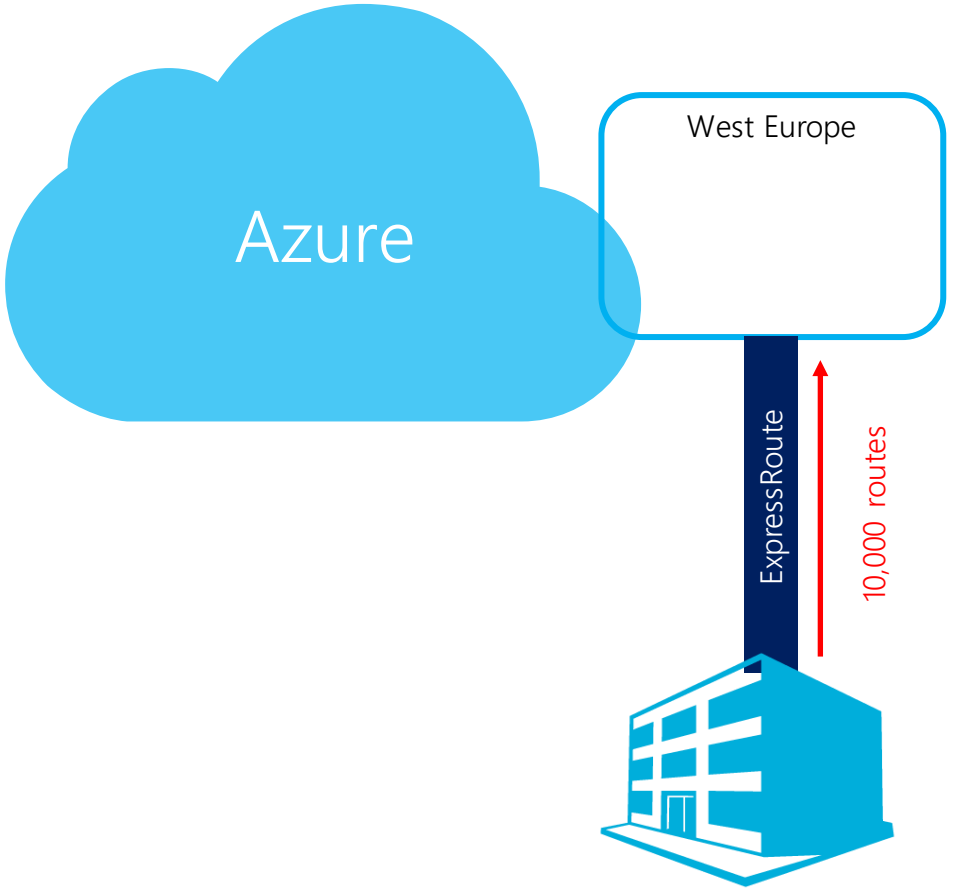
An ExpressRoute connection allows access to other regions within the same ‘geo-political area’.



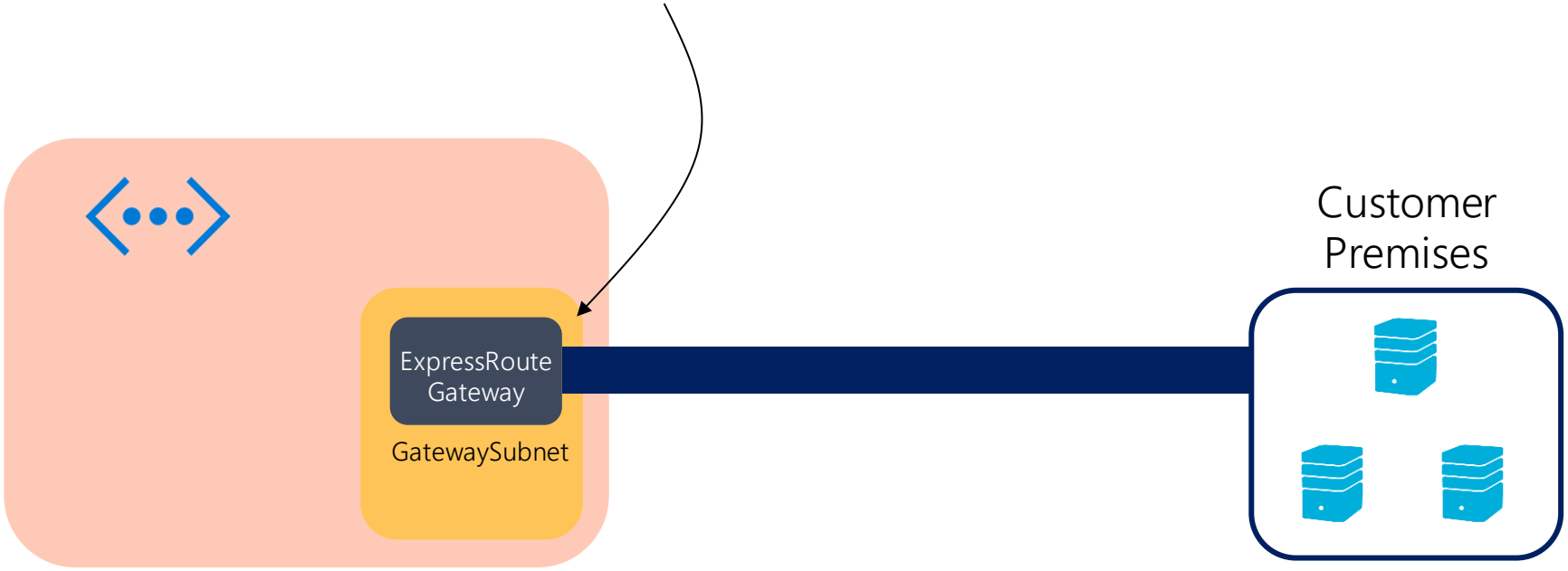
The ExpressRoute ‘Premium’ add-on enables connectivity across geo-political boundaries.



The ‘Premium’ add-on also gives you increased route limits (10,000).

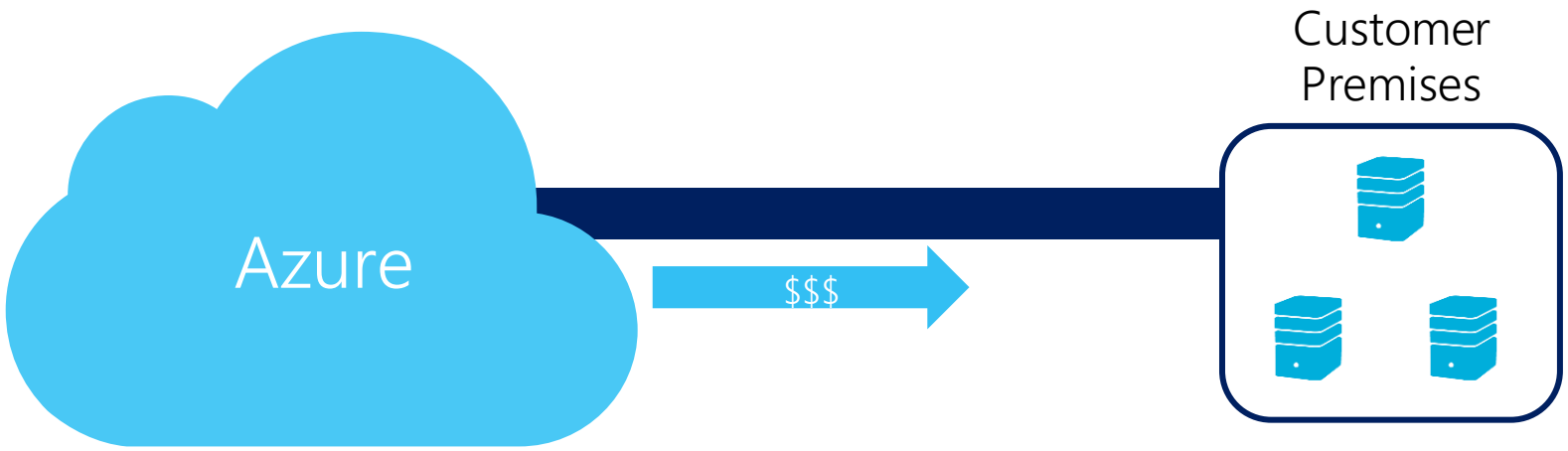


ExpressRoute uses a gateway that resides in the GatewaySubnet (similar to VPN gateways).

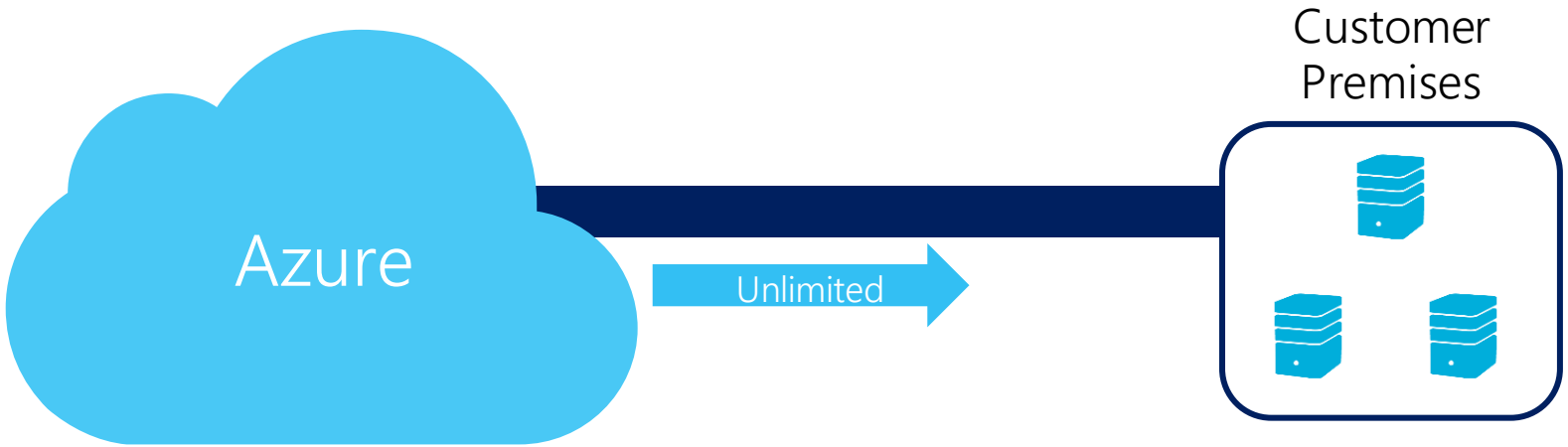


ExpressRoute has two types of billing: *Metered* and *Unmetered*.

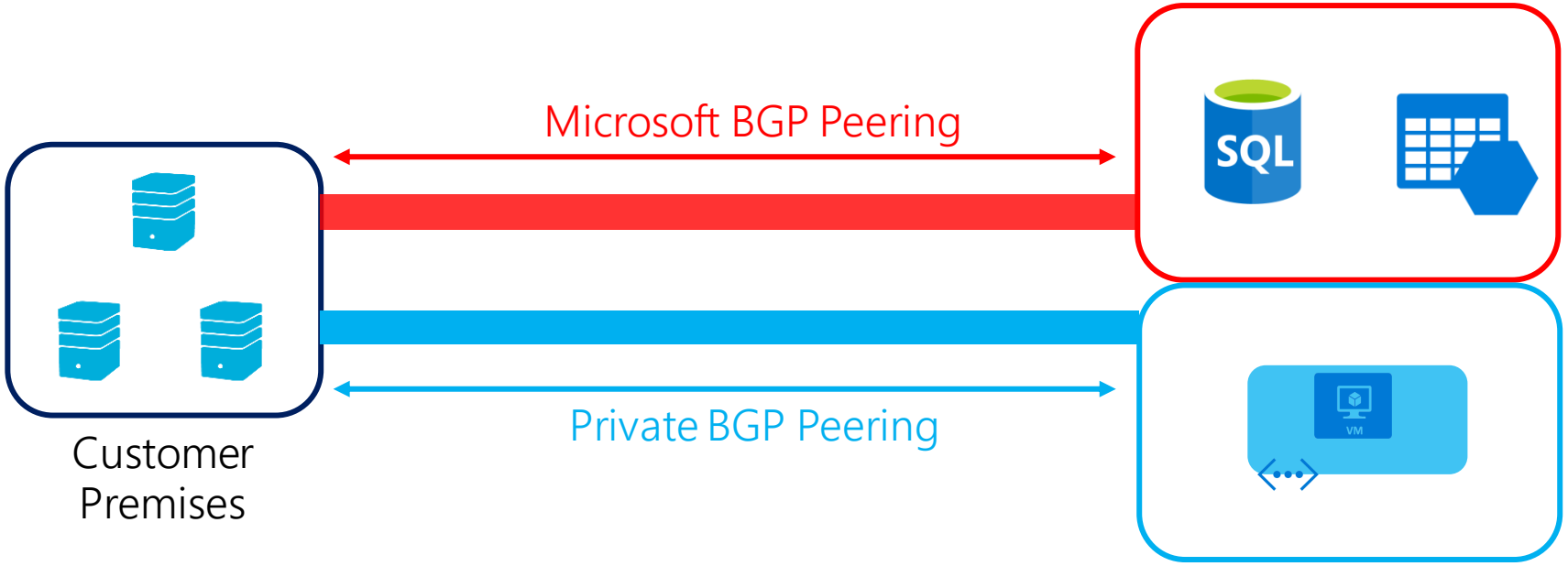
With Metered billing, you pay for outbound data transfers (per GB).



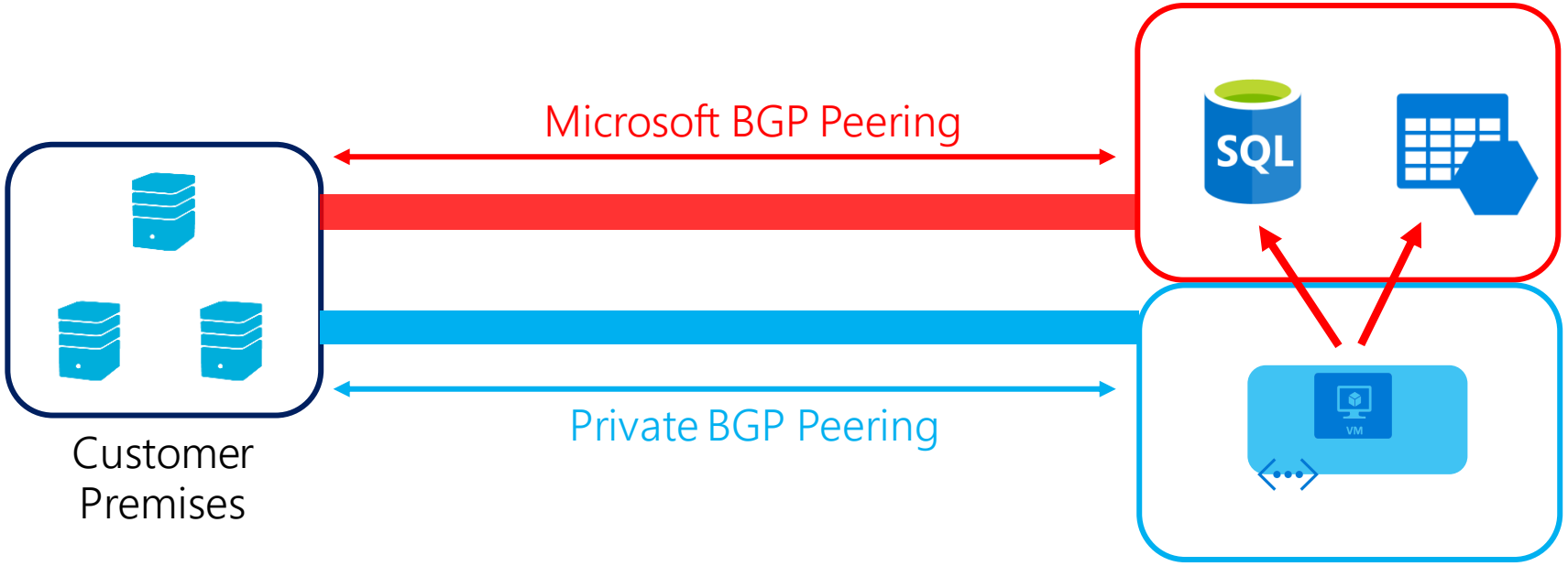
Unmetered billing gives you unlimited outbound data transfer (but of course is more expensive in the first place).



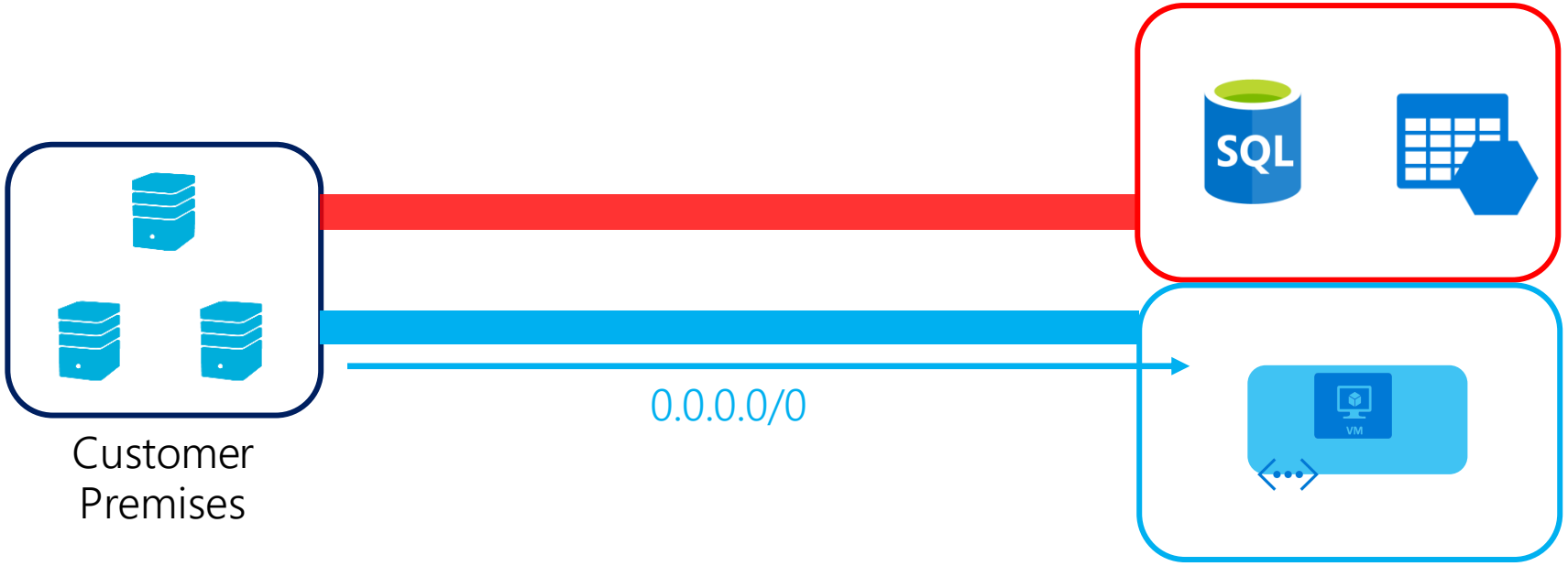
Let’s imagine a scenario where our Azure VMs need to communicate with public Azure services (e.g. Azure Storage or Azure SQL).



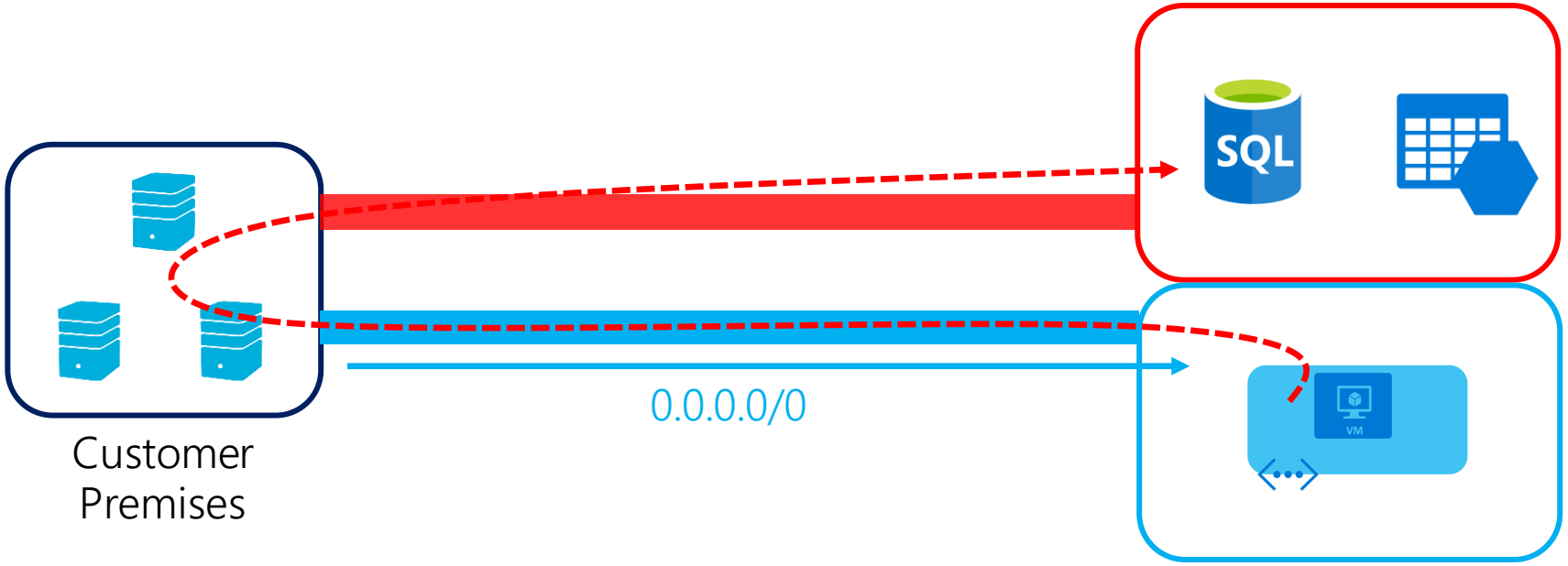
In most circumstances, traffic will be kept on the Microsoft network (i.e. it won't traverse the ExpressRoute circuit).



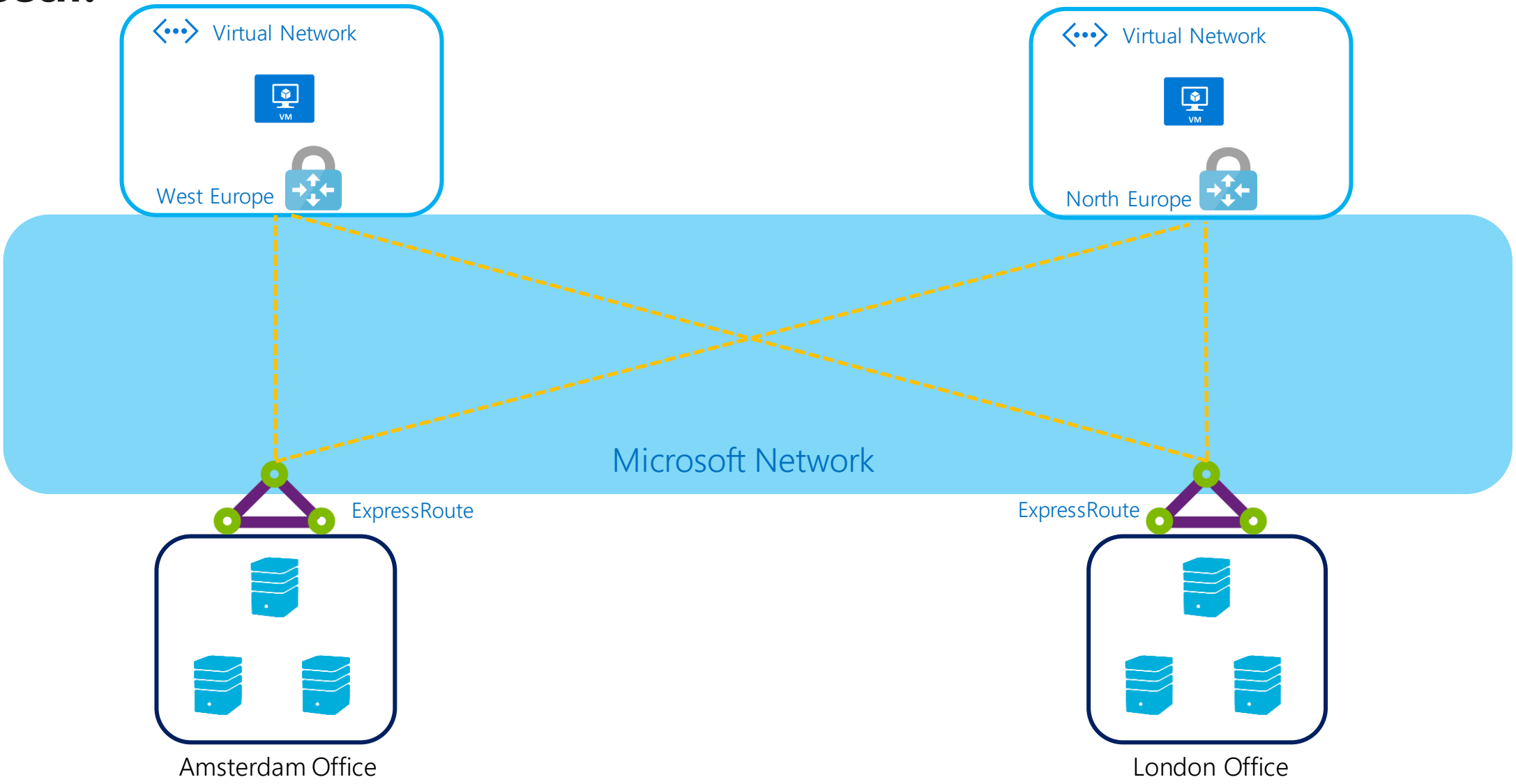
However, some customers might choose to advertise a default route to Azure via the private peering (to force all traffic via on-premises).



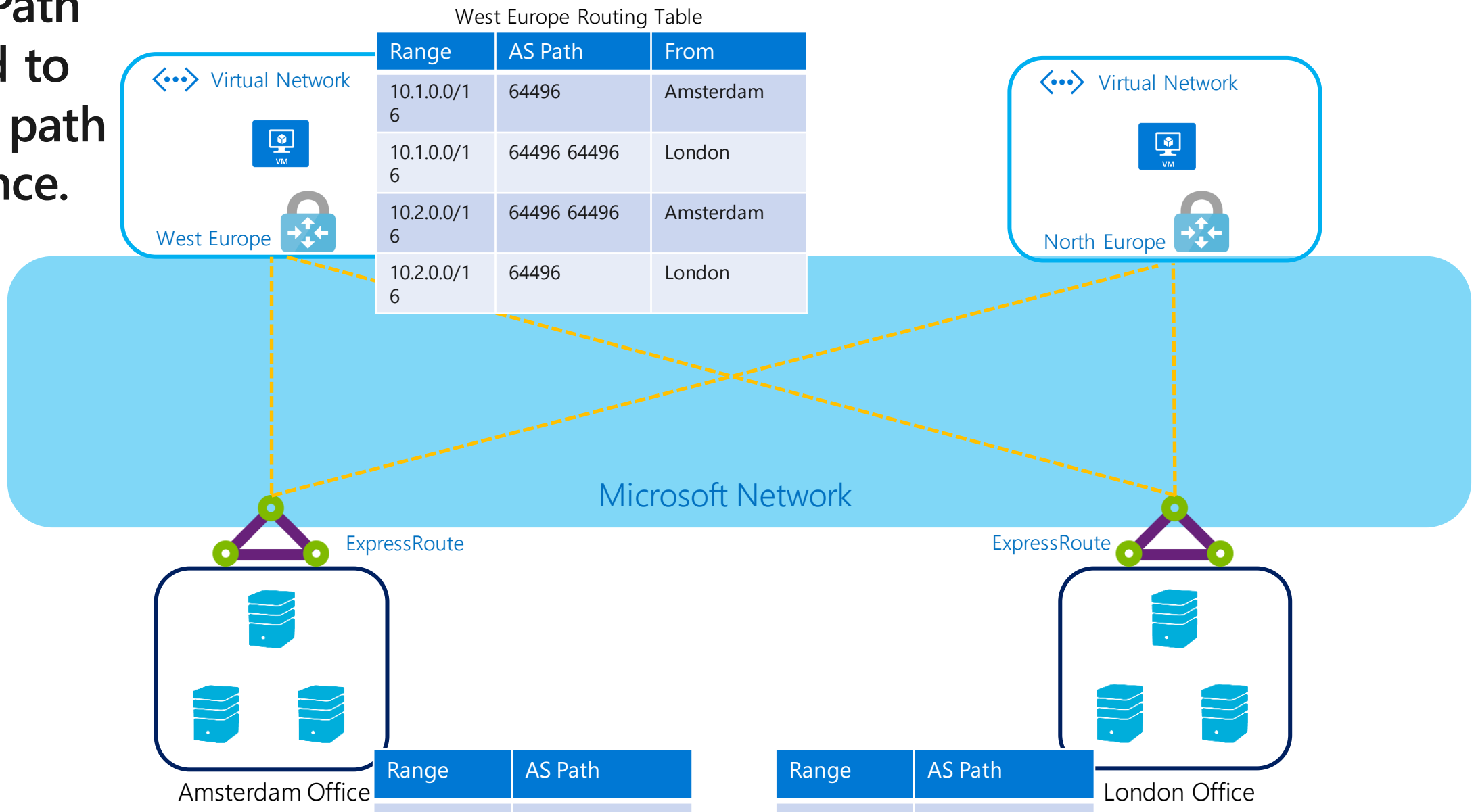
In that case, all traffic between your VNet and Azure public services will *hairpin via on-premises*.



In scenarios with multiple ER circuits, asymmetric routing can occur.



Use AS Path Prepend to help with path preference.



Range	AS Path	From
10.1.0.0/16	64496	Amsterdam
10.1.0.0/16	64496 64496	London
10.2.0.0/16	64496 64496	Amsterdam
10.2.0.0/16	64496	London

Range	AS Path
10.1.0.0/16	64496
10.2.0.0/16	64496 64496

Range	AS Path
10.1.0.0/16	64496 64496
10.2.0.0/16	64496

Dos and don’ts

Do use public ASN for Microsoft Peering

Do have more than one ExpressRoute at different locations for redundancy

Do enable connectivity on both primary connection and secondary connection for high availability

Do use URL-based filtering for secure access to Microsoft clouds

Do not configure default route (0.0.0.0/0) in the UDR on ExpressRoute gateway subnet

Do not deny the Internet by setting the default route (0.0.0.0/0) in the NSG on ExpressRoute gateway subnet

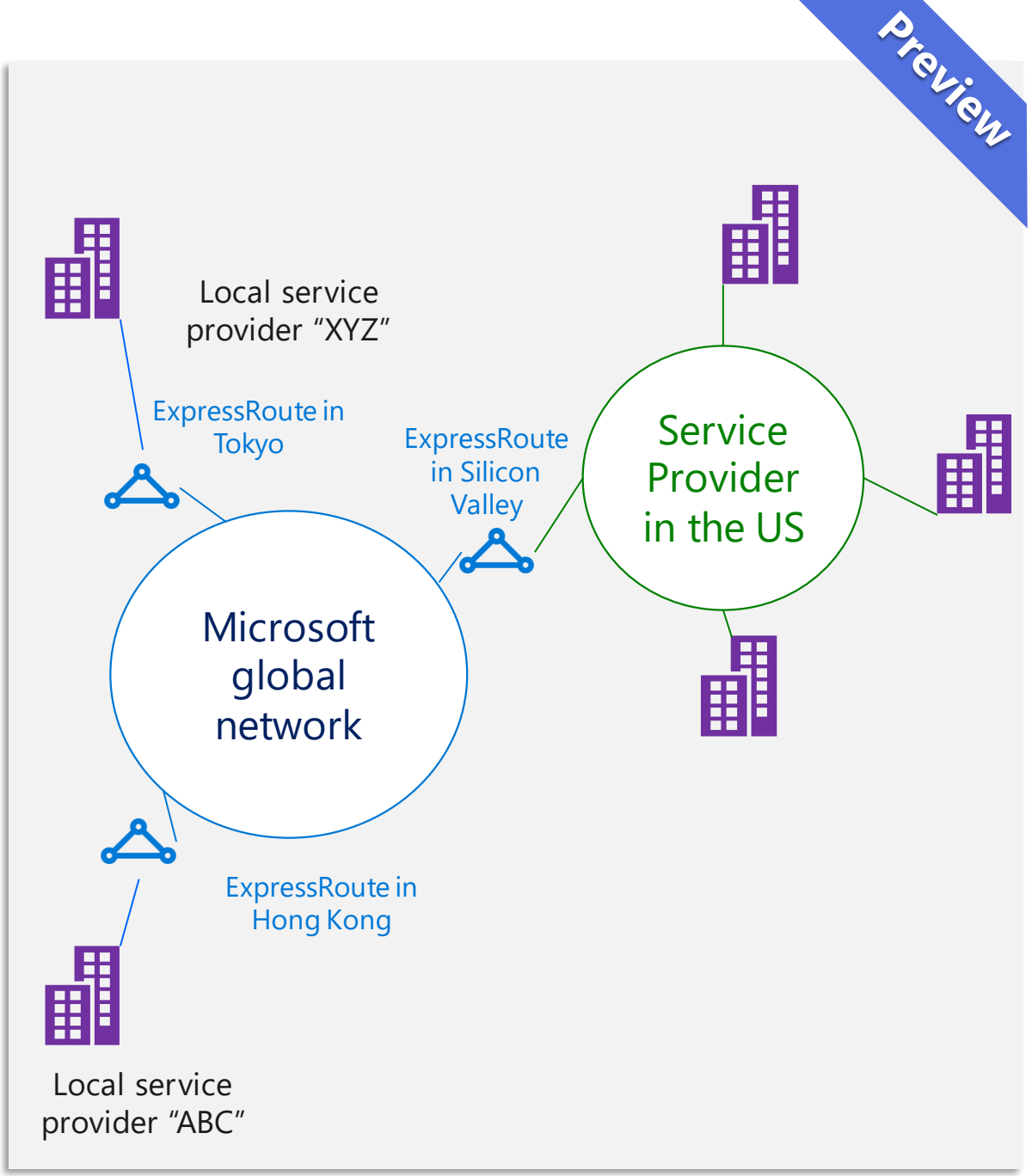
Do not advertise the same prefix over Microsoft Peering and Internet

ExpressRoute Global Reach

On-demand site-to-site connectivity using ExpressRoute

Customers can use existing ExpressRoute circuits
Traffic stays on Microsoft's global network
Complements service provider's WAN solution
Supported on Standard and Premium circuits
Extends Azure private peering
Capability for full mesh network
Available in Public Cloud
- Australia, Canada, France, Hong Kong, Ireland, Korea, Japan, Netherlands, U.S., U.K.,
- Singapore coming soon

Available in Azure US Government



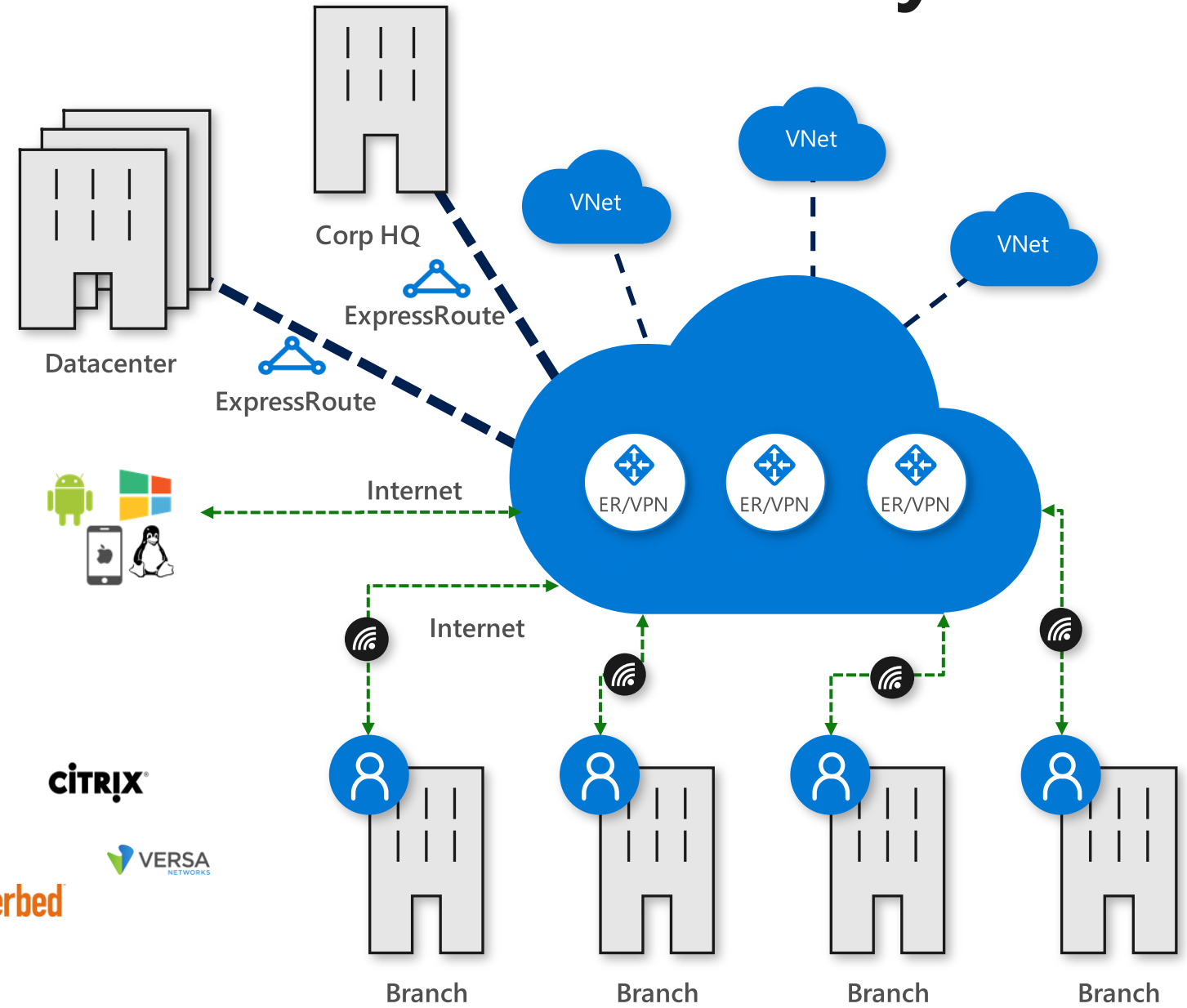
Azure Virtual WAN – Unified Cloud Connectivity

Overview

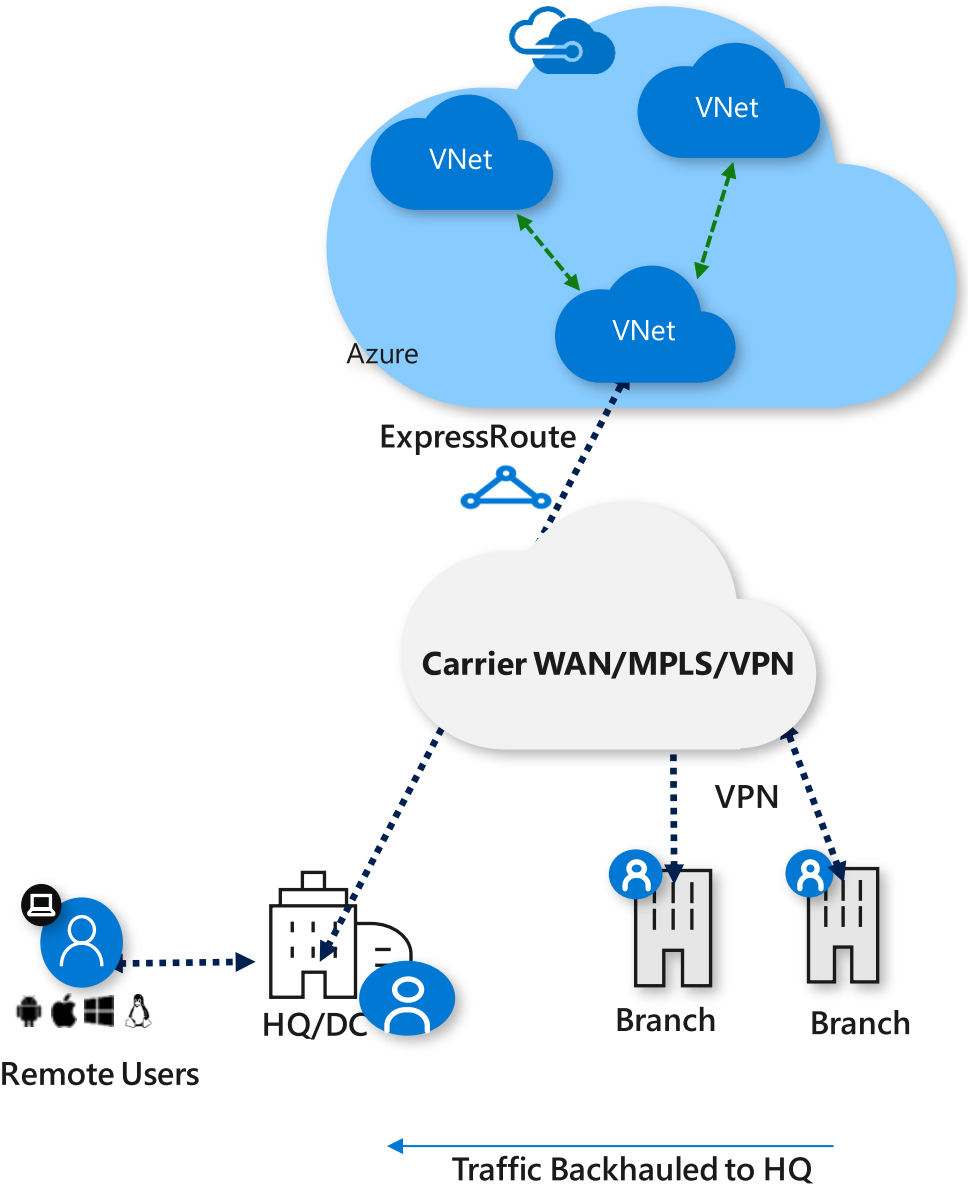
- Large scale Internet VPN for branch offices
- ExpressRoute for datacenters / HQ
- P2S VPN for mobile workers
- Local Internet breakout for O365
- Ecosystem of SDWAN/VPN partners

Preview features

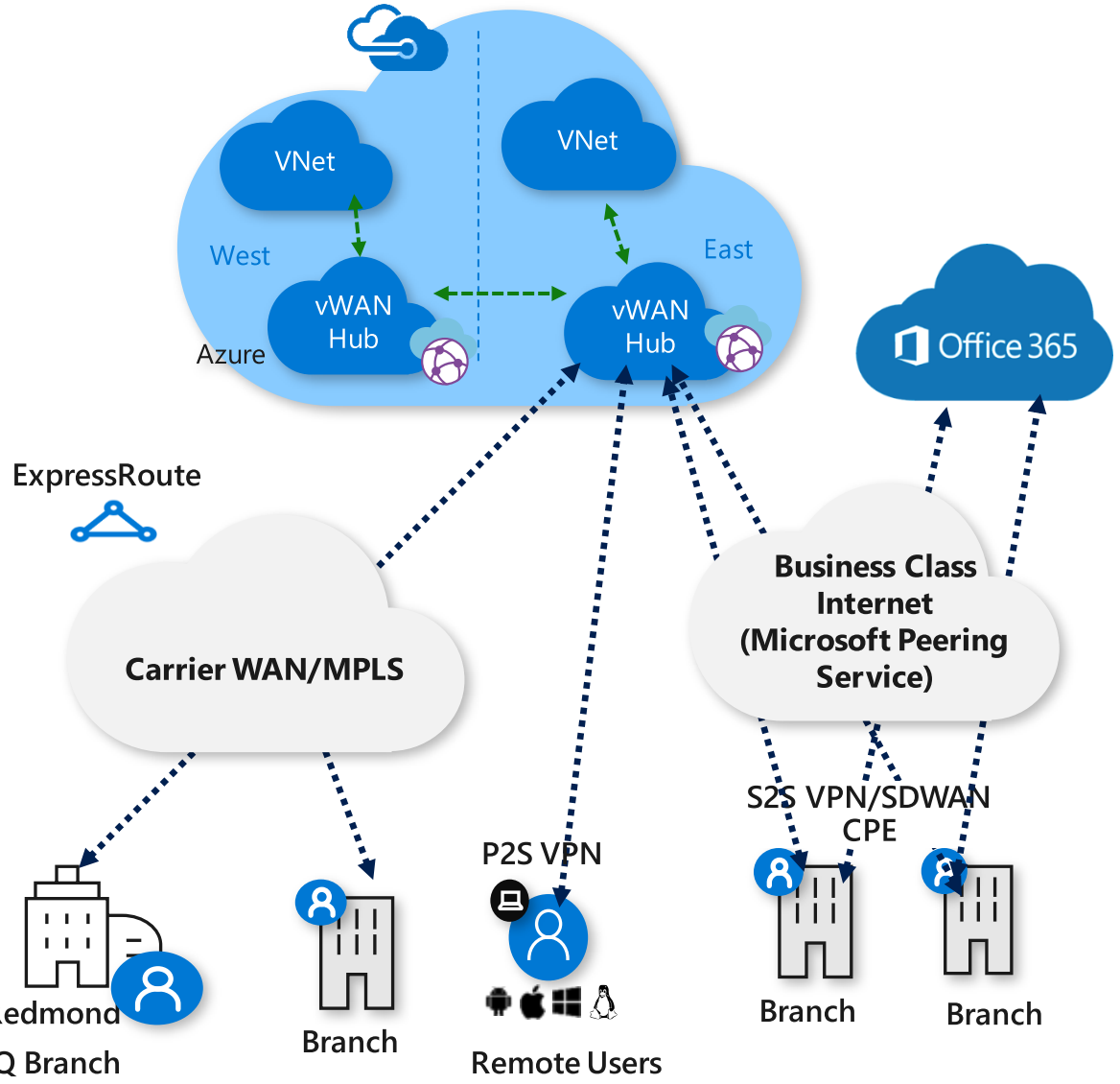
- ExpressRoute
- P2S with IKEv2 and OpenVPN
- O365 Policy



Traditional WAN



Azure Virtual WAN

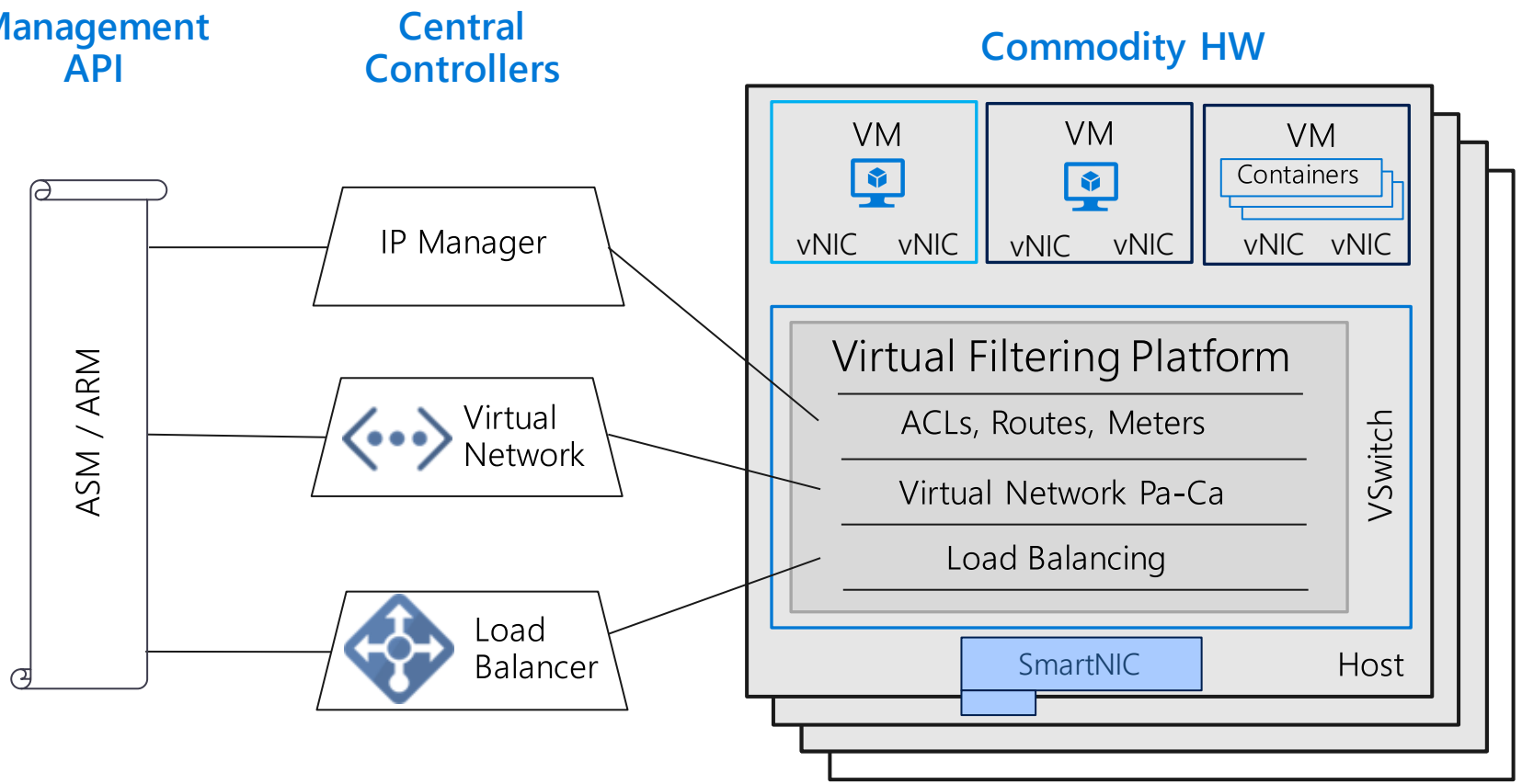


Microsoft Confidential

Software Defined Networking (SDN)

Azure SDN
Basis of all NW virtualization in our datacenters

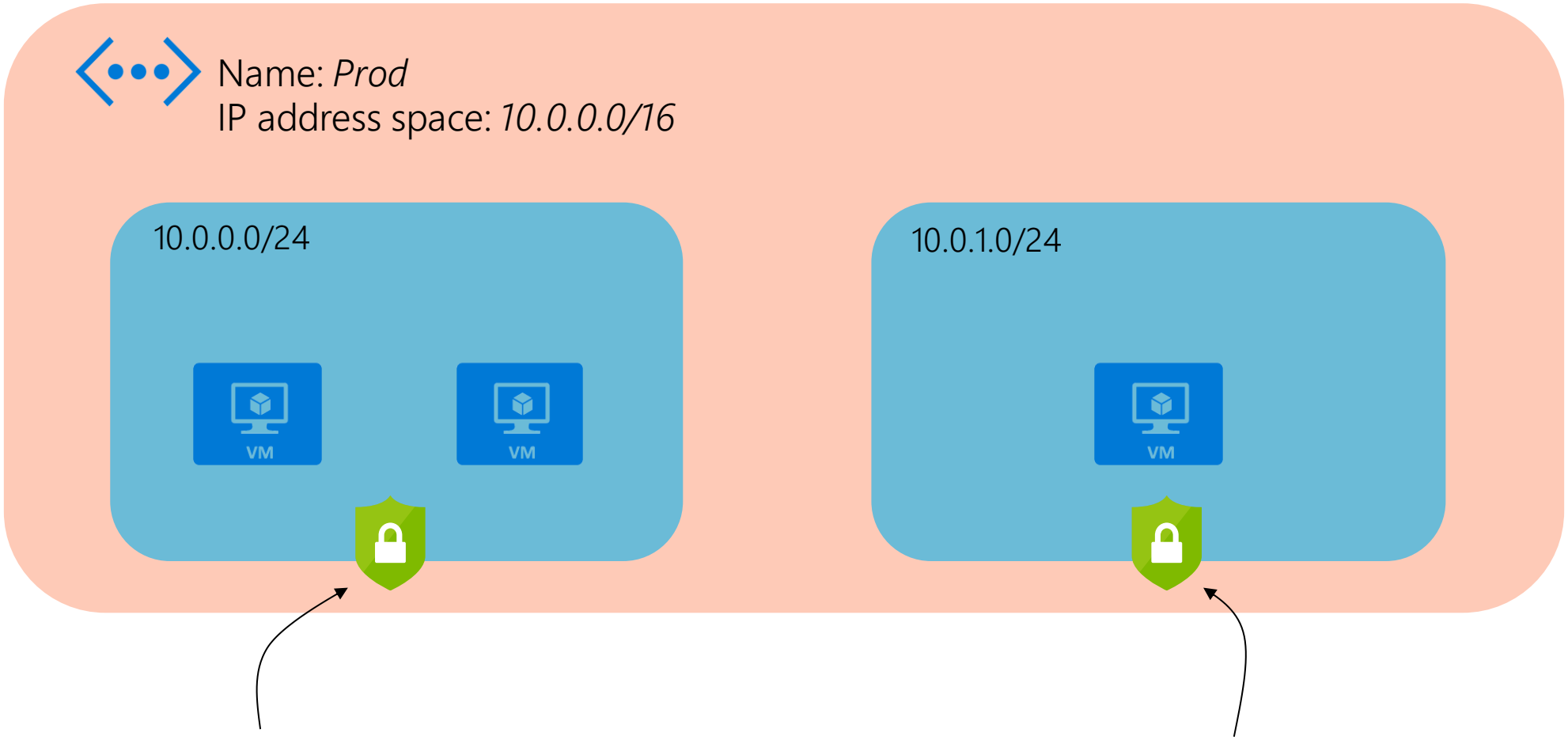
Decoupled
SDN allows compute to evolve and converge to a single allocator



Key to flexibility and scale is Host SDN

Network Security Groups

Network Security Groups (NSGs) lock down access to a subnet or VM.



An NSG is essentially a list of access control rules, permitting or denying traffic based on various criteria.

Source IP	Dest IP	Protocol	Port	Action
10.1.1.2	30.1.1.5	TCP	22	Allow
10.1.1.2	*	TCP	80	Allow
10.1.1.2	30.1.1.6	TCP	443	Allow
*	*	*	*	Deny

NSGs contain a number of default rules.

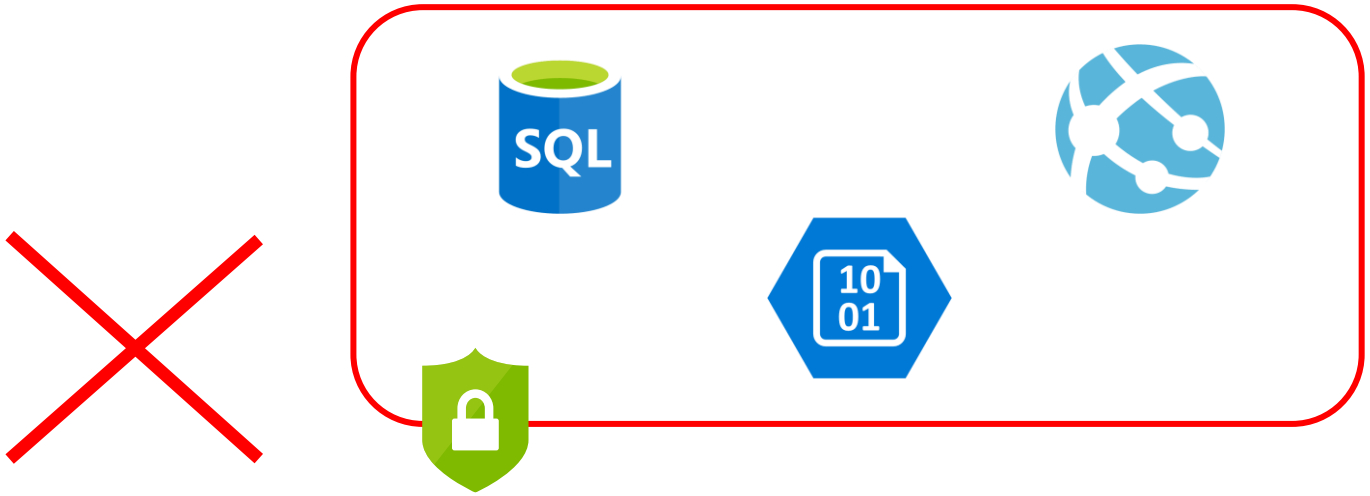
Inbound

Name	Priority	Source IP	Destination IP
AllowVNetInBound	65000	VirtualNetwork	VirtualNetwork
AllowAzureLoadBalancerInBound	65001	AzureLoadBalancer	*
DenyAllInBound	65500	*	*

Outbound

Name	Priority	Source IP	Destination IP
AllowVNetOutbound	65000	VirtualNetwork	VirtualNetwork
AllowInternetOutBound	65001	*	Internet
DenyAllInBound	65500	*	*

NSGs *only* work if a resource is connected to a VNet – they do not work for other resources (e.g. PaaS services).



Simplified Security Group Management

From the Internet

Within the VNet

Within Azure

Network Security Groups (NSG)

Protects your workloads with distributed IP ACLs

Attach: Subnet and NICs

Simplified configuration with augmented security rules

Application Security Groups (ASG)

Micro-segmentation for dynamic workloads

Named monikers for groups of VMs

Removes management of IP addresses

Service Tags

Named monikers for Azure service IPs

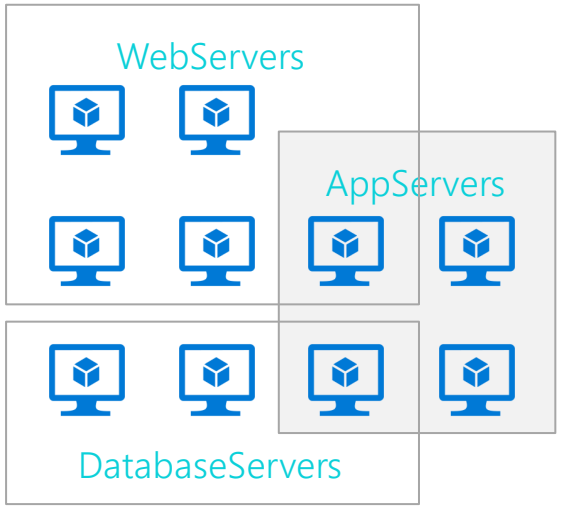
Many services tagged including AzureCloud

Logging and troubleshooting

NSG flow logs for traffic monitoring

Integrated with Network Watcher

JIT access policies with Azure Security Center



NSG

	Name	Source	Destination	Port
Allow	AllowInternetToWeb	Internet	WebServers	80,8080 (HTTP)
Allow	AllowAppToOnPrem	AppServers	10.10.128.0/22, 10.20.36.0/20, 192.168.65.0/20, 192.168.10.0/24	22, (SSH) 21, (FTP) 3389, (RDP) 3306 (MySQL)
Allow	AllowAppToExternalAPI	AppServers	148.234.0.0/16, 190.22.33.8/30	443 (HTTPS)
Allow	AllowDBServerToStorage	DatabaseServers	Storage	Any
Deny	DenyAll	Any	Any	Any

VNet Integration

Azure services—integration patterns

Service deploys dedicated instances into customer's VNet

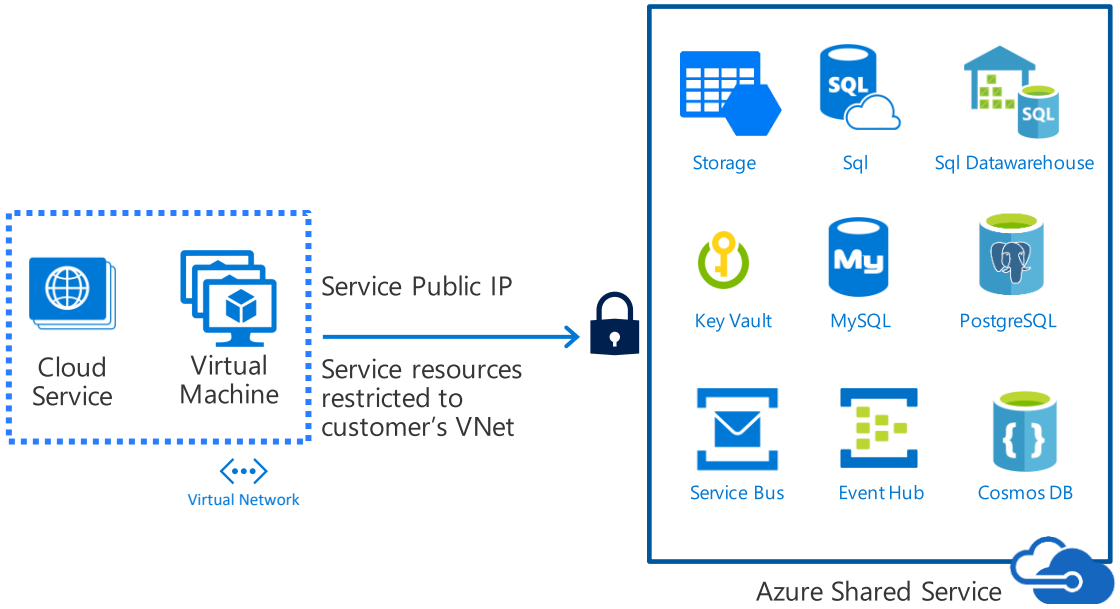
We call it **VNet Injection**

GA; available today

Inbound/outbound access to Azure Service; Single Tenancy

Extend VNets to multi-tenant service

We call it **VNet Service Endpoints**



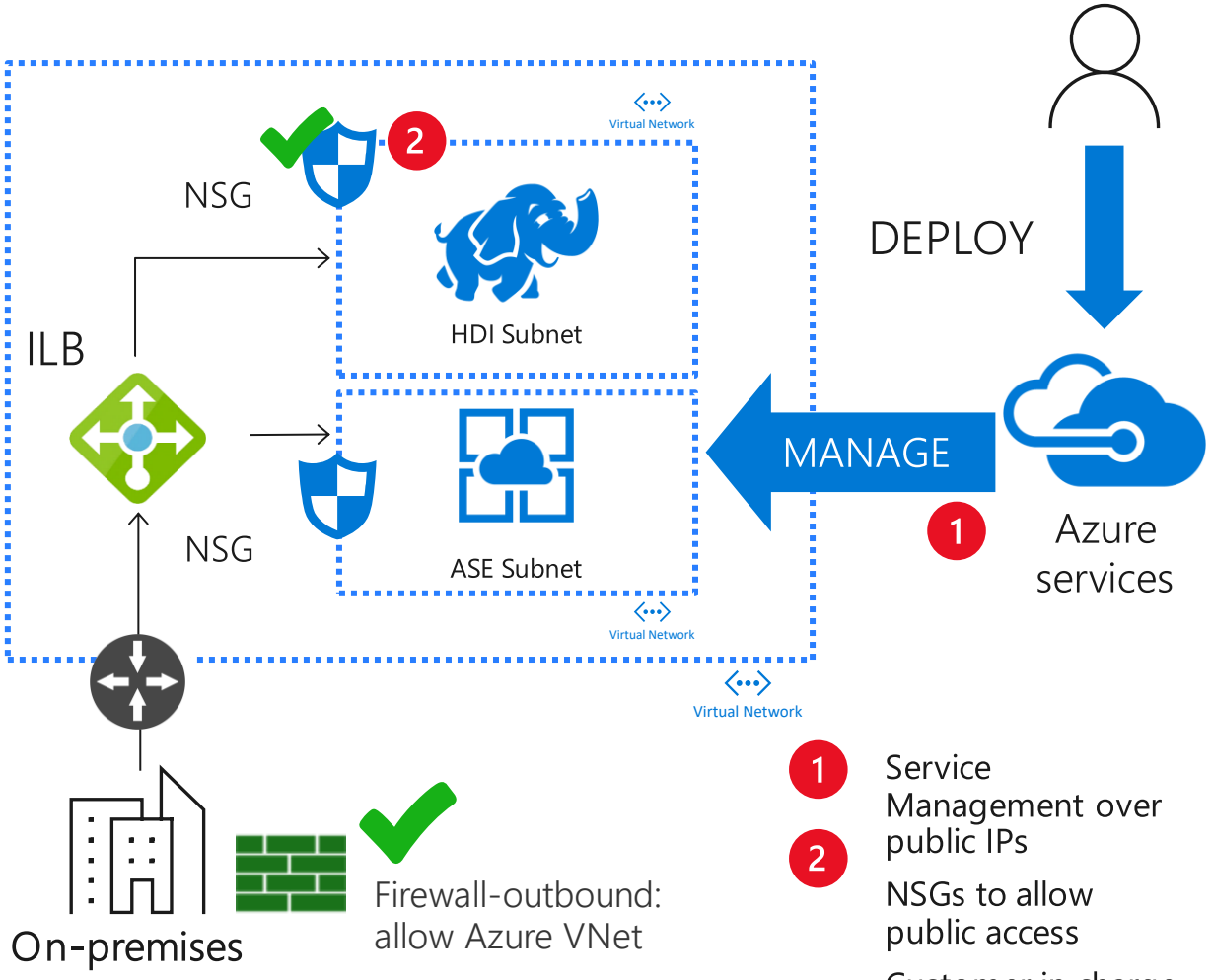
GA; available today

Outbound-only access to Azure Service; Multi Tenancy

VNet injection

Service deploys dedicated instances into customer's VNet

- Services in your VNet, managed by Azure!
- Single Tenancy; Private IPs for service resources
- Service data plane exposed privately, ILB
- Inbound and Outbound access to Azure Service
- On-premises access through Site-to-Site or ER private peering



1. Service Management over public IPs
2. NSGs to allow public access

Customer in charge of NSGs/UDRs

VNet injection: Challenges

1. Service management happens over public IP. Not completely private
2. NSGs to allow Public IPs for Management traffic

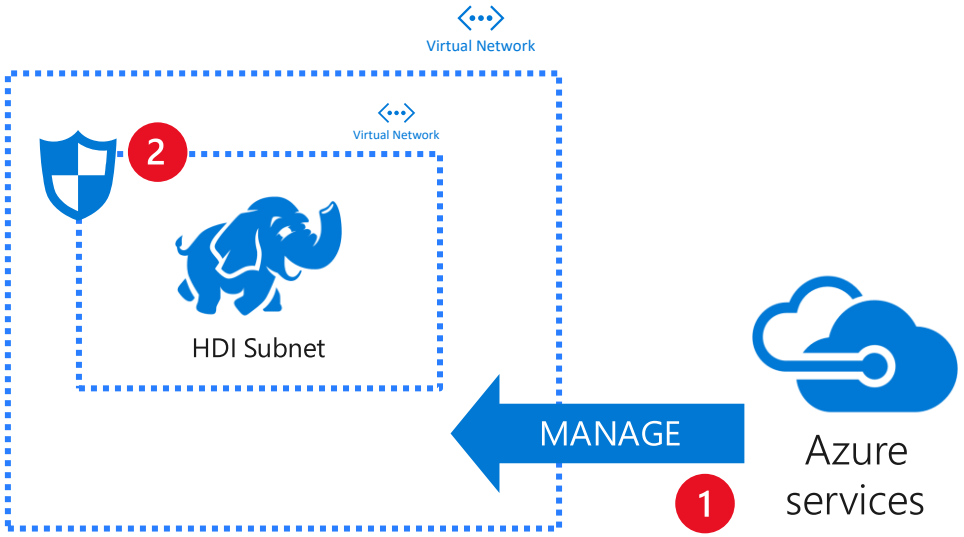
✓ Accidental NSGs and UDRs changes can cause management traffic* failures

Customer wants complete isolation!!

*Management traffic is needed for actions like monitoring the health of the instances and provide required scale based on load

**NSG—Network Security Groups

***UDR—User Defined Routes



Network Security Group			
Action	**Direction**	**Name**	**Source/Destination**
Allow	Inbound	Management	Service Management IP Addresses
Allow	Outbound	Management	Service Management IP Addresses
Allow	Outbound	InternetDependencies	List of Services Tags

Route table		
Name	**Address Prefix**	**Next Hop**
Management	*Service Management IP Addresses*	Internet
Internet dependency	List of Services Tags List explicit IP addresses	Internet

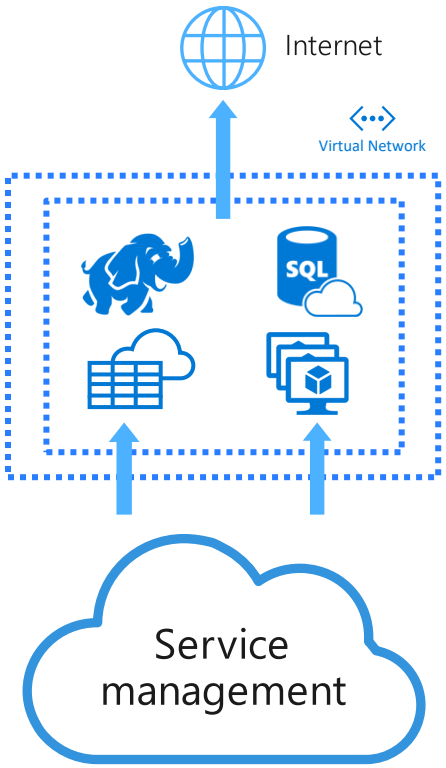
VNet injection: Mitigations

Available Today

- ✓ Service Tags for ease of managing IPs and limiting exposure

Roadmap

- ✓ Preconfigure NSGs and UDRs on the subnet where PaaS service is injected*
- ✓ Isolate management traffic from data traffic
  - Management traffic not visible to customer or mapped into private space
  - Customer expressed policies (NSG, UDR) does not apply to management traffic



Network Security Group

Action	Direction	Name	Source/destination
Allow	Inbound	Management	*ServiceManagementTag*
Allow	Outbound	Management	*ServiceManagementTag*

Route table

Name	Address prefix	Next hop
Management	*ServiceManagementTag*	Internet

Network Security Group

Action	Direction	Name	Source/destination
~~Allow~~	~~Inbound~~	~~Management~~	~~*ServiceManagementTag*~~
~~Allow~~	~~Outbound~~	~~Management~~	~~*ServiceManagementTag*~~

Route table

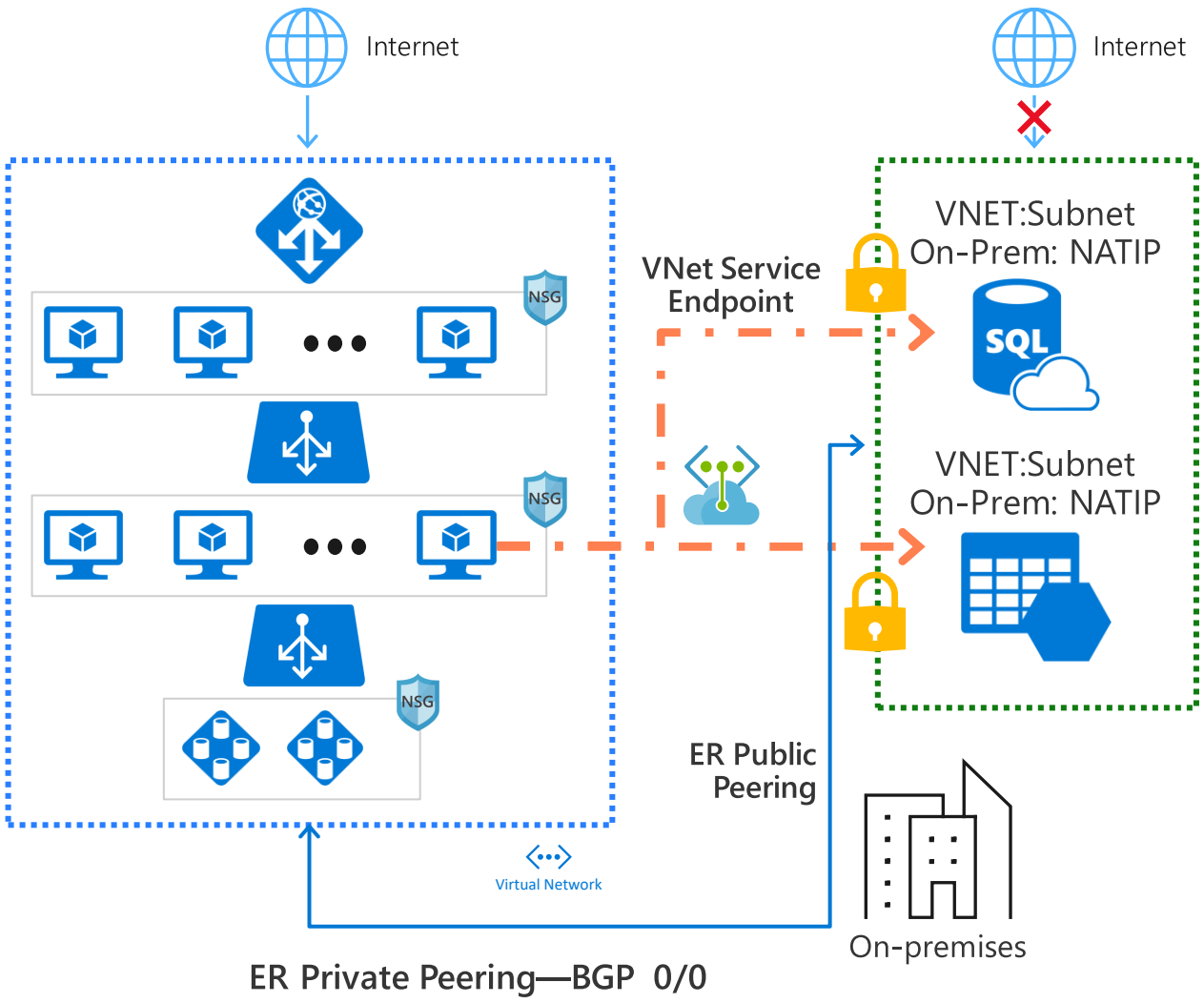
Name	Address prefix	Next hop
~~Management~~	~~*ServiceManagementTag*~~	~~Internet~~

*We call it Delegation and Network Intent Policies

VNet service endpoints

Shared resources secured to customer's VNet

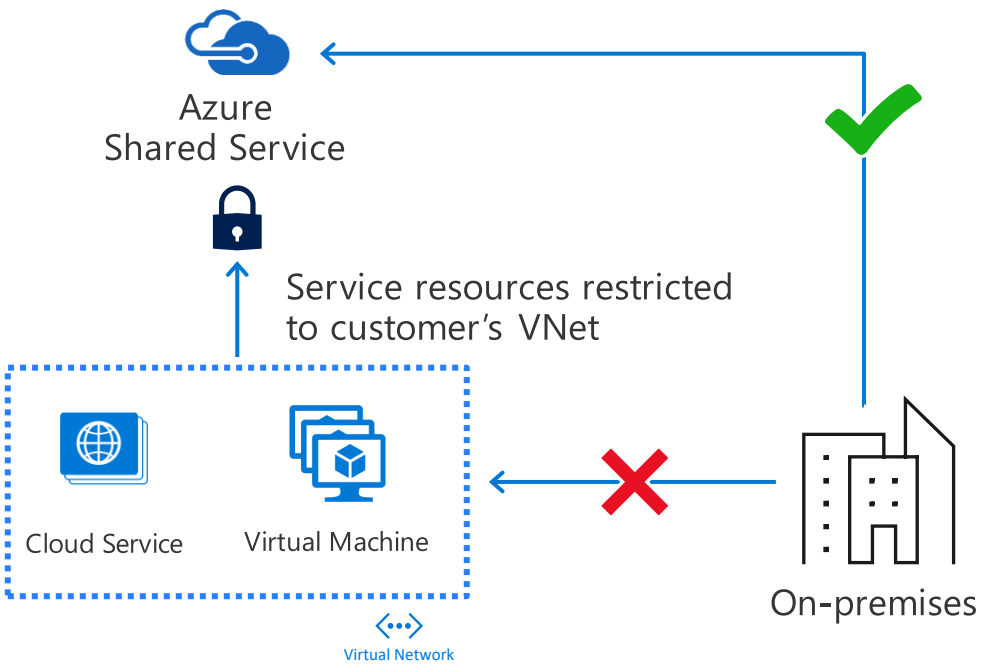
- Directly extends VNet to the service
- Secure critical Azure resources to only your VNet
- Traffic remains on the Microsoft backbone
- On-premises access through ER public peering
- Forced Tunneling overridden



VNet service endpoints: Challenges

- No data exfiltration Protection*
- No Transitive Access**
- No On-Prem access through ER or VPN gateway
- NSG Open to Public IPs and DNS entries Public
- No Support for Force-Tunneling***

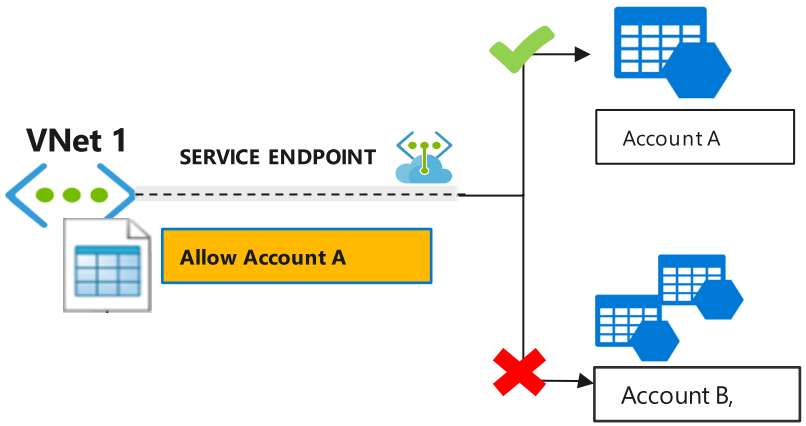
*Ability to restrict VNet access to specific Service accounts
**Service outside VNet lose access to Services locked to VNet
***Ability to take traffic back to on-prem for inspection



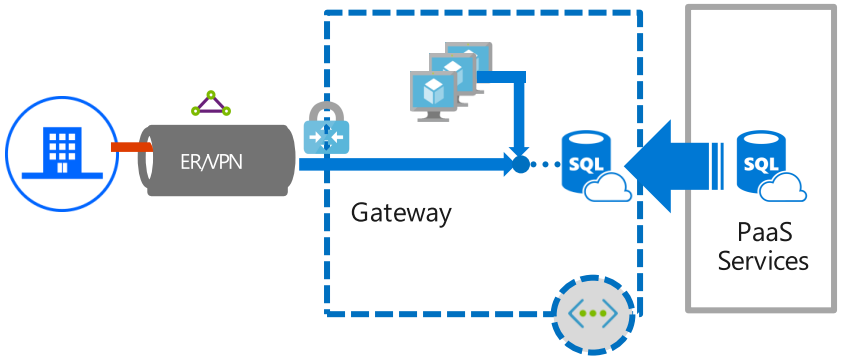
Network Security Group				
Action	Name	Source	Destination	Port
Allow	*Azure Storage*	*Virtual Network*	*Azure.Storage*	*Any*
Allow	*Azure SQL*	*Virtual Network*	*Azure.SQL*	*Any*

VNet service endpoints: Mitigations

- Available Today
  - ✓ Provide ability to restrict VNet access to specific service accounts*
  - ✓ Transitive access through storage bypass and MSI**
- Roadmap
  - ✓ Map Azure service resources to customers VNet***
    - Provides data exfiltration protection
    - No NSGs to Public IPs and DNS entries are Private
    - Support for Force-Tunneling
    - Access from On-Prem through ER Private Peering



SERVICE ENDPOINT POLICIES
(Preview)



PRIVATE LINK

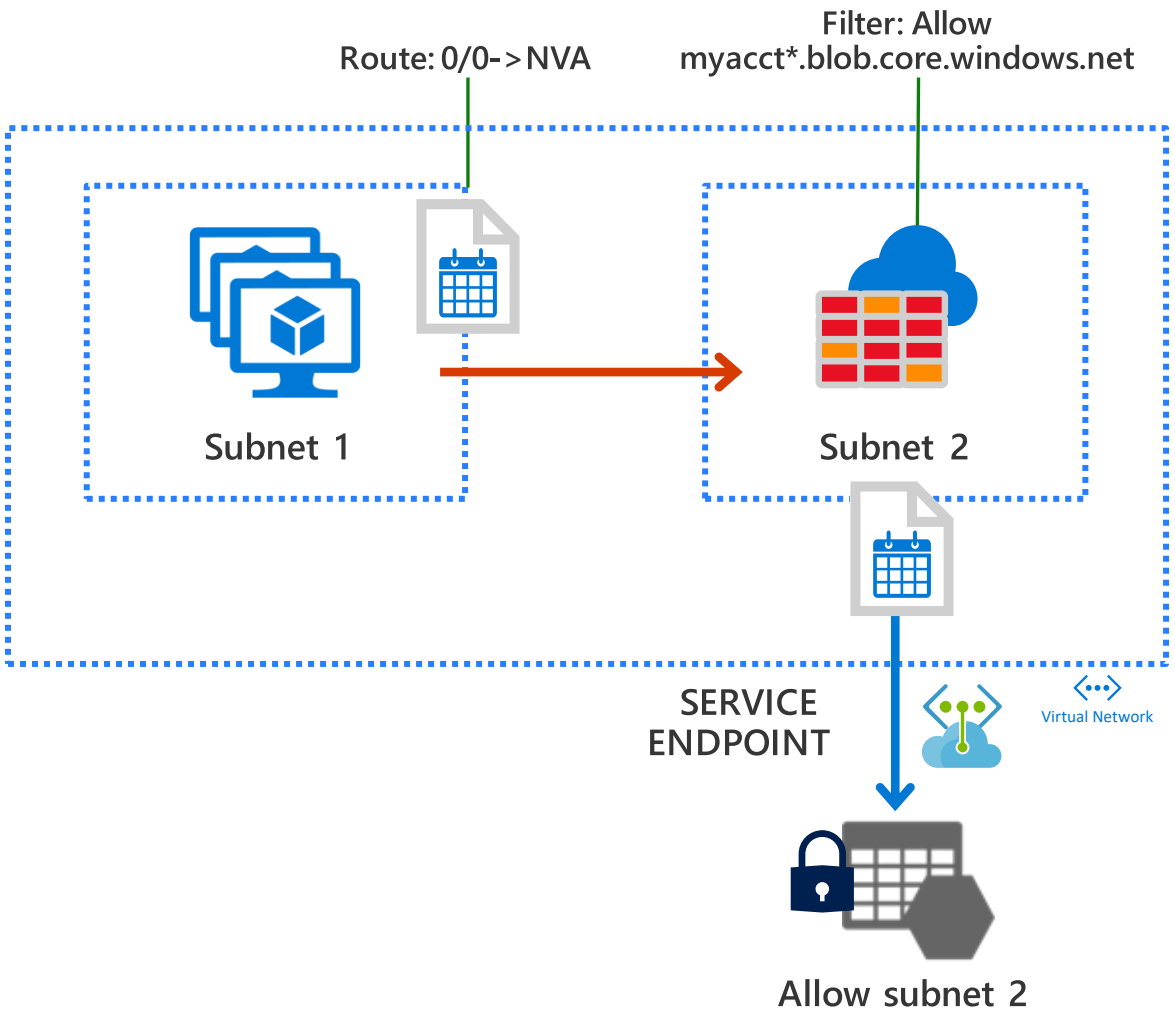
* Service Endpoint Policies (Storage Preview in West Central US & West US 2)

**Azure services support Managed Service Instance (MSI) to allow other services to talk to it on customers behalf

***Private Link (codename)

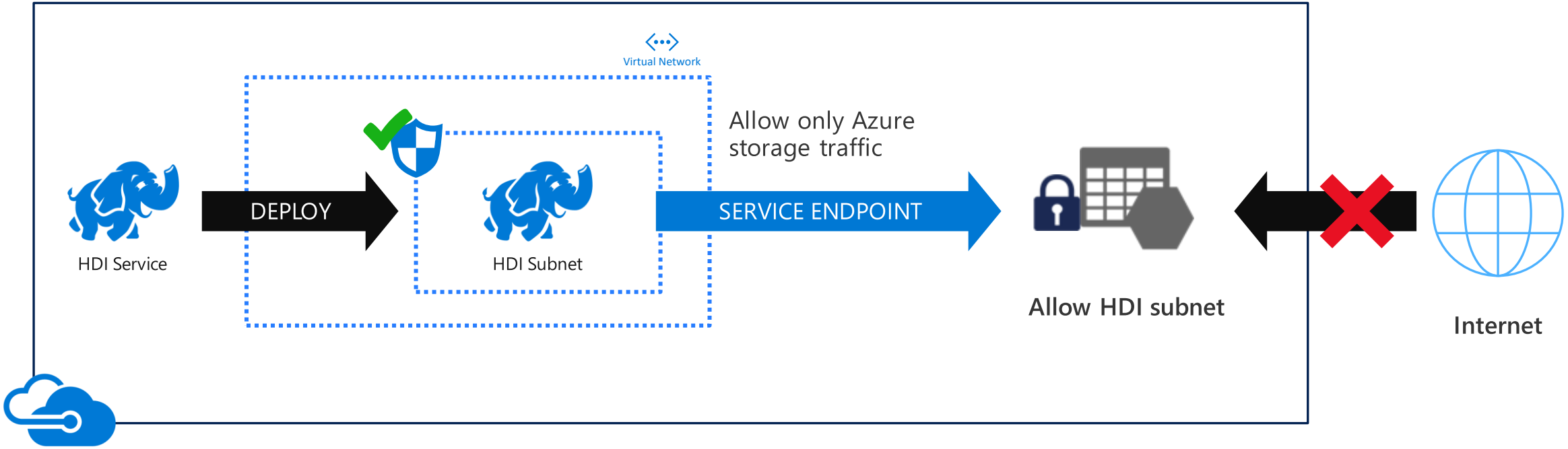
VNet service endpoints: filter service traffic

- Optionally, you can use NVA/ Azure Firewall or auditing/filtering service traffic
- Service endpoints bypass NVA/ Azure Firewalls for service traffic, if set on originating subnet
- Set endpoint on NVA/Firewall subnet instead and secure resources to that subnet
- Note: Service endpoint policy shouldn't be enabled on Azure Firewall subnet.



Stitching Azure services together

- Secure Azure resources to managed service subnets with endpoints
- Applies to all services directly deployed into VNet



Private Link and Private Endpoints

Private Link

Highly secure and private connectivity solution for Azure Platform

Simplified usability experience

- Unified and consistent experience across Azure Services, partner services and your own services

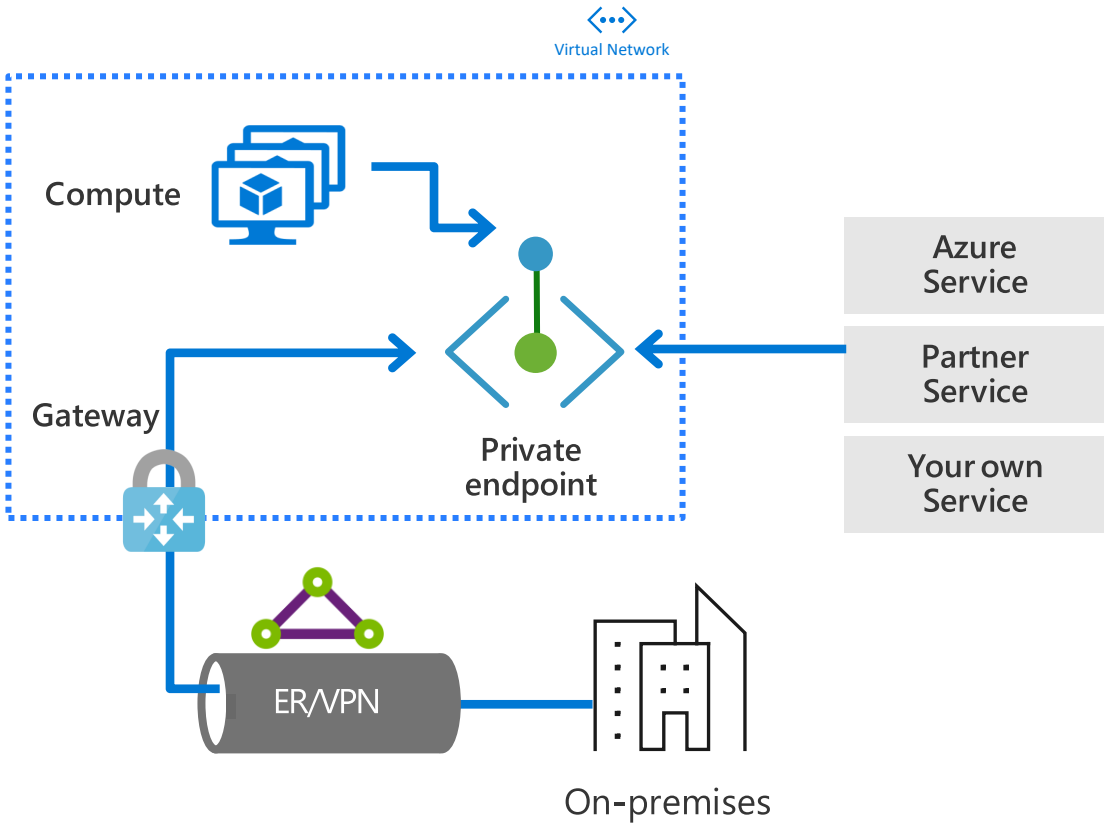
Simplified networking

- No internet gateway, NAT device, public IP address, ER, or VPN connection to communicate with the service
- Access from Peered and On-prem Networks privately. Use ER private peering or VPN tunnels, removing internet traffic
- Predictable IP addresses for PaaS resources

Simplified security

- Routing over private network. Meet compliance needs
- NSG and firewall configuration clean within customer address space

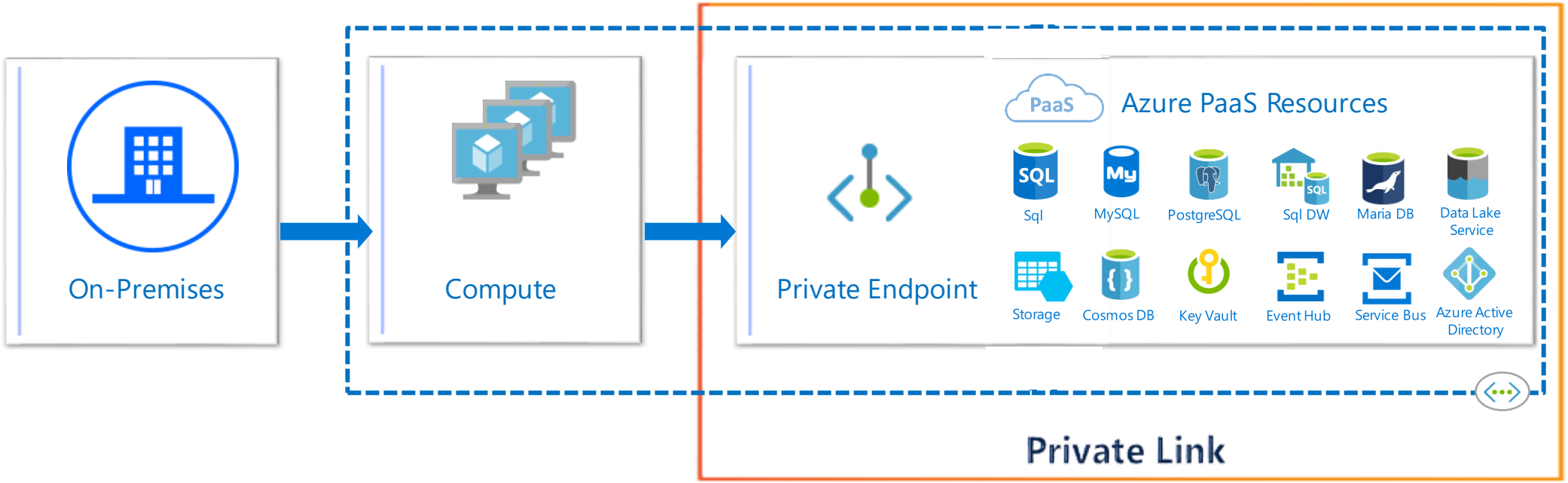
Privately consume services in your VNET



Private Preview Available for SQLDB; Your Own Service
Roadmap services – Storage, SQL DB/DW, CosmosDB

Private Link for Azure Services

Privately connect to PaaS services using Virtual Networks



Connect privately

- ✓ From on premises using ER private peering or VPN, removing internet traffic.
- ✓ From your Virtual Networks including peering

PaaS Resources

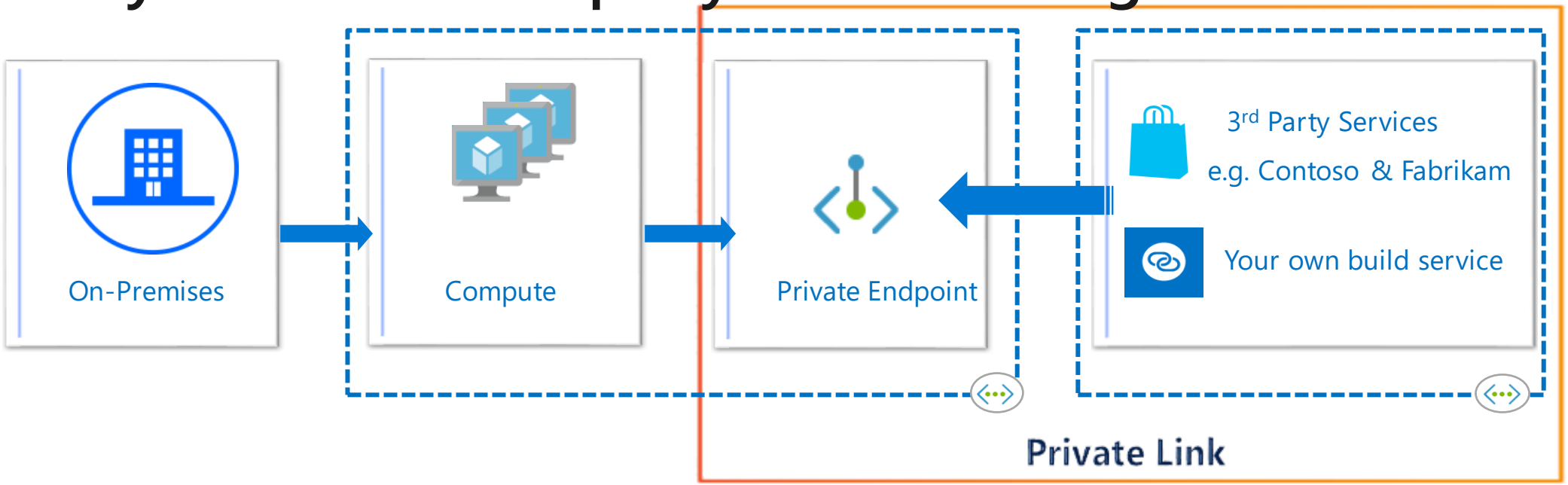
- ✓ Specific resource deployment
- ✓ Cross subscription and AD support with approval workflow
- ✓ Connect globally with PaaS resources on different regions

Security simplified

- ✓ NSG & Firewall configuration clean within customer address space
- ✓ Micro-segmentation with ASGs
- ✓ Predictable IP addresses for PaaS resources

Private Link for 3rd Party Services

Privately connect to 3rd party services using Virtual Networks



Connect privately

- ✓ From on premises using ER private peering or VPN, removing internet traffic
- ✓ From your Virtual Networks including peering
- ✓ Familiar security configuration with NSG & ASGs

3rd Party Service

- ✓ Consume services from service providers
- ✓ Cross AD support with approval workflow
- ✓ Connect globally with services on different regions

Your Own Service

- ✓ Build your own services and connect privately
- ✓ Overlapping IP addresses handled by the platform

Azure DDoS Protection

DDoS attack types

Volumetric attacks

Brute force attacks that saturate network links and resources

Example attacks

TCP Flood, Excessive TCP segments, Spoofing, Amplification etc.

Protocol attacks

Attacks that consumes actual server resources, or those of intermediate communication equipment, such as firewalls and load balancers

Example attacks

GRE, ICMP ping flood, excessive fragments, land attack etc.

Resource attacks

Target the application layer and involves triggering a back-end process to overwhelm resources on the target system

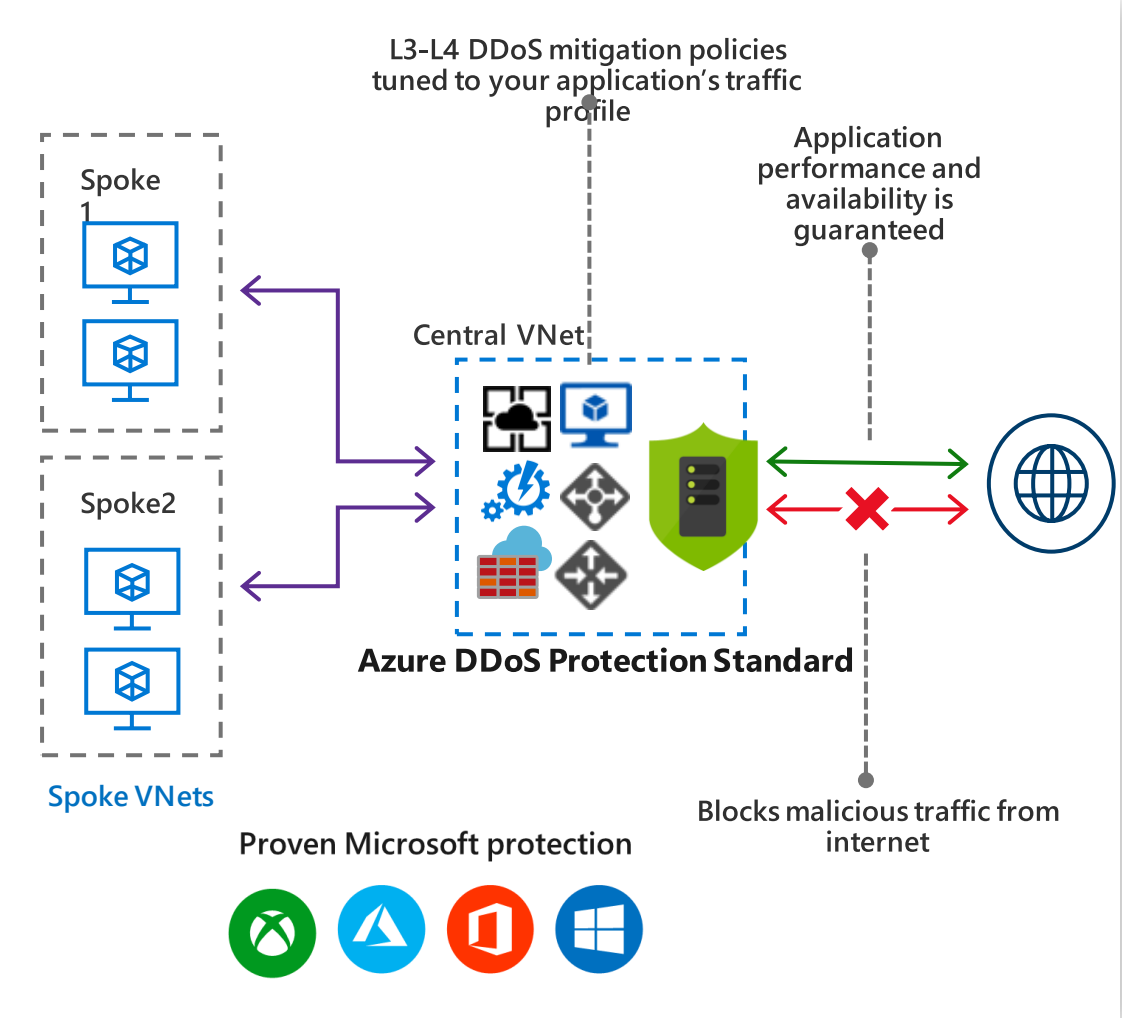
Example attacks

Slowloris, Poison cache, Slow post, etc.

Azure DDoS Protection

Cloud scale DDoS protection tuned to applications

- **Designed into the global network**
- **Global distribution of the attack during large scale attacks**
  - 25+ Tbps global mitigation capacity
- **Simple to provision** for all your virtual network resources
- **Always on monitoring** with near real time telemetry and alerting
- **Automatic** network layer attack mitigation
- **Protection policies** tuned to your application's traffic profile
- **DDoS Attack Analytics**
  - Near real time network attack mitigation flow logs
  - Attack data snapshots and full post attack summary
- **DDoS Rapid Response**
  - Specialized Rapid Response team support during active attacks
  - Custom mitigation policy configuration
- **SLA guarantee and cost protection**
  - Protection against the unforeseen costs of a DDoS attack



Choosing VMs for Networking Performance

Right size:

- The largest size of the newest SKUs have the highest allocated bandwidth
- But more CPUs isn’t always better for throughput
  - Scheduling overhead
  - Competing workloads

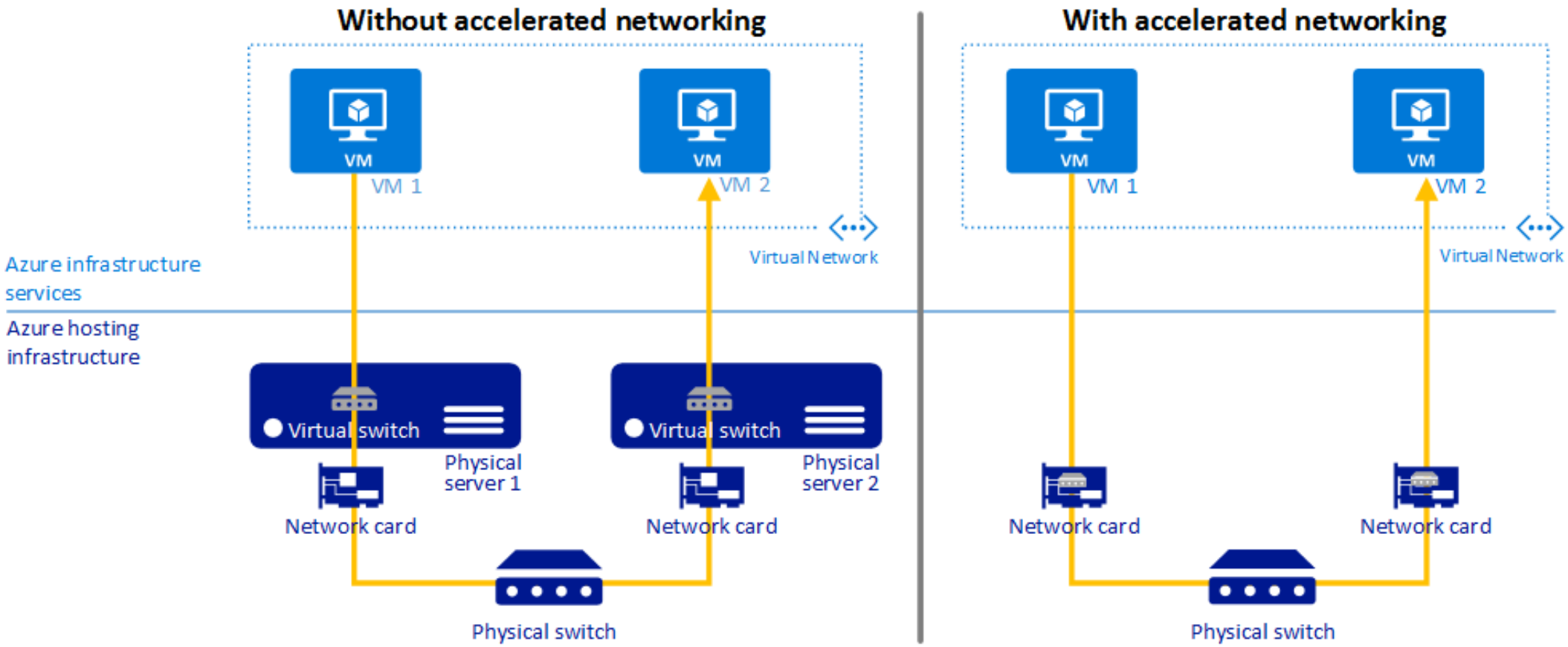
Right series: Blend performance properties and cost

- Networking and cost vs. CPU, Memory, Storage
- Cost for low-to-high Gbps: F-series, Dv2, especially D15v2
- Cost for highest Gbps: Dv3, Ev3, Fv2
- Storage optimal with AccelNet: Lv2
- Max mem: M128 and variants of the same size

Accelerated Networking Internals – SR-IOV

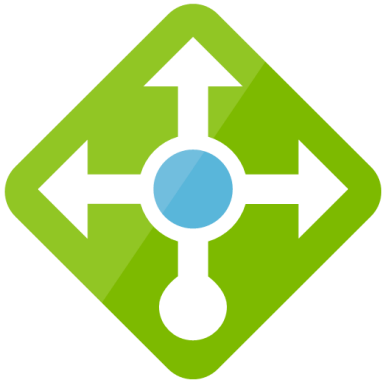
SDN/Networking policy applied in software in the host

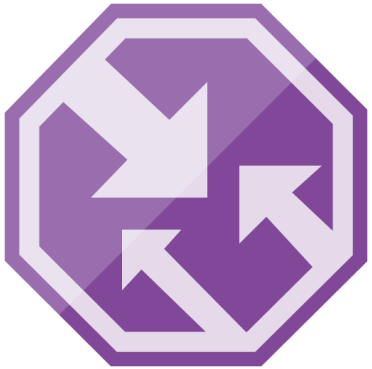
FPGA acceleration used to apply all policies



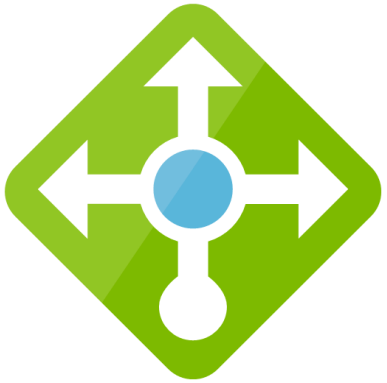
Load Balancing

Azure has a few different types of load balancer available.

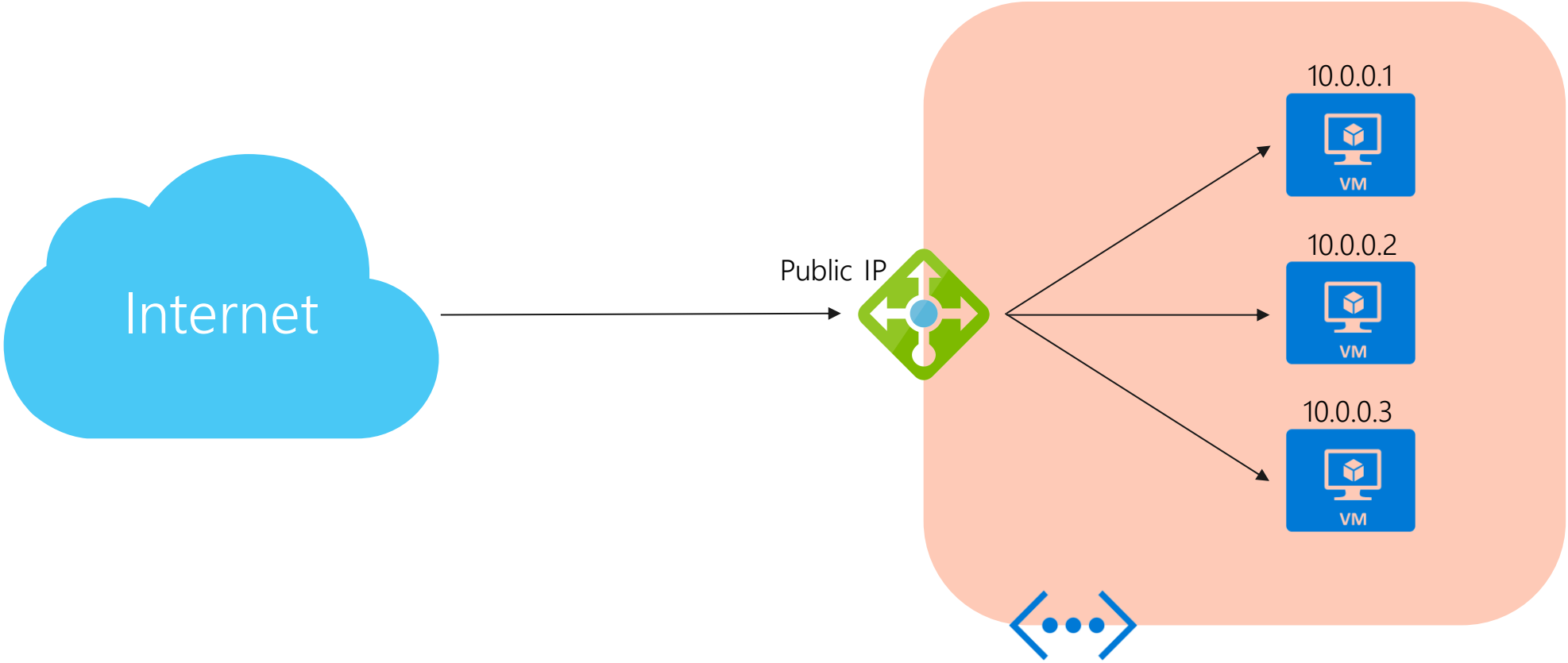




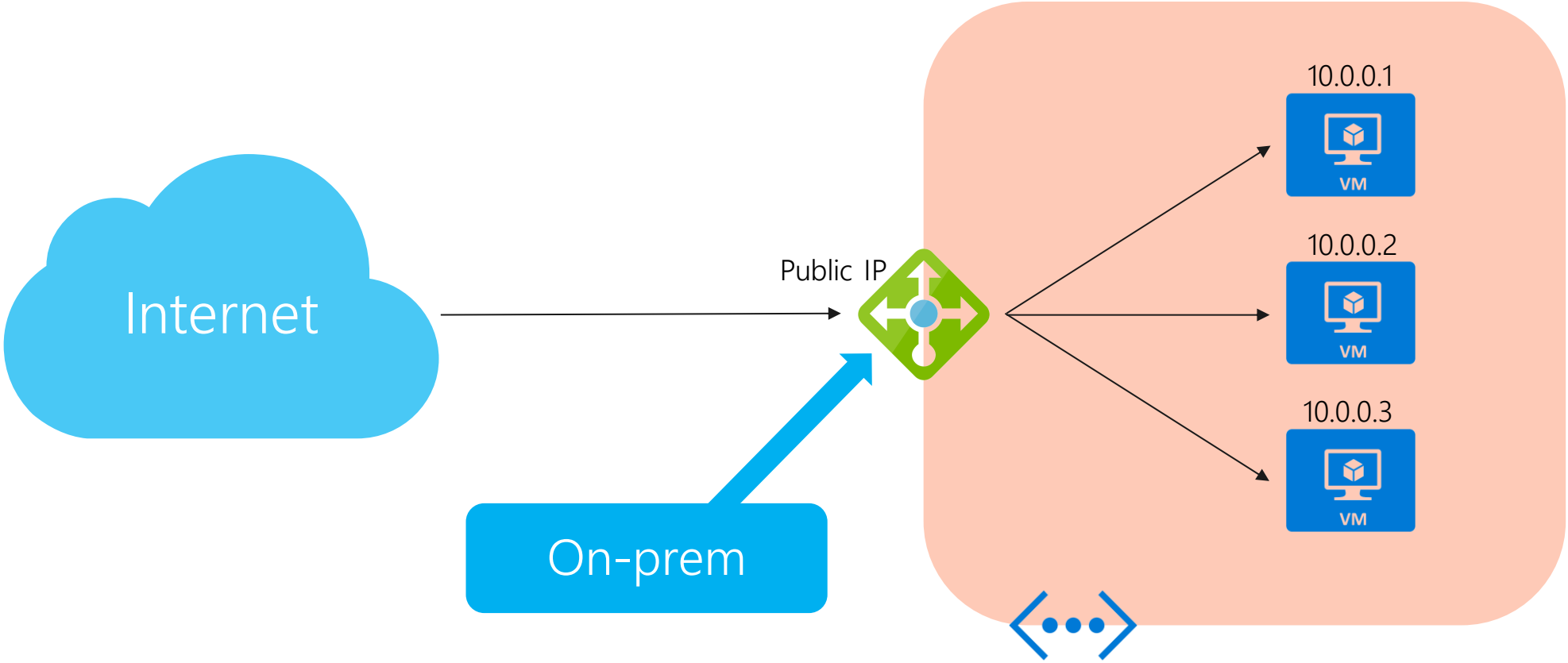
First, let’s look at the basic / standard load balancer.



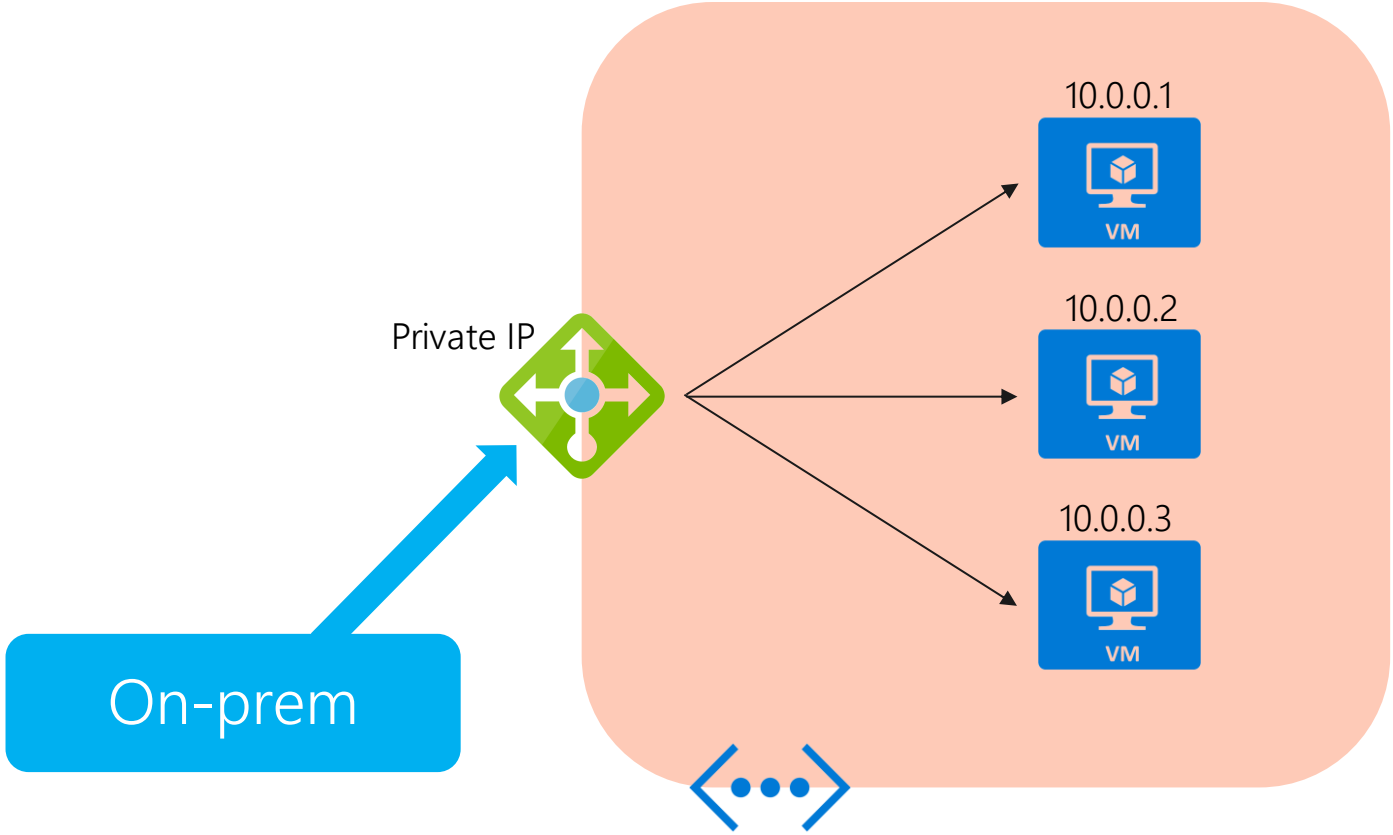
The Azure load balancer offers basic L4 load balancing.



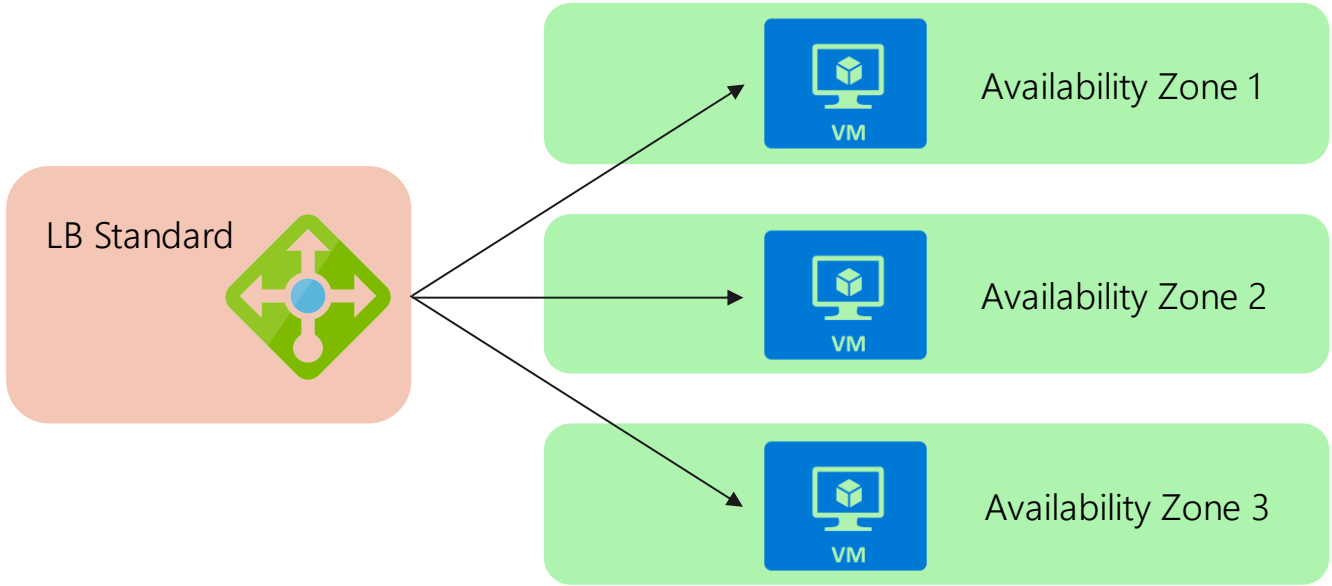
Load balancers can operate in ‘public’ mode (i.e. with a public IP address)



Load balancers can also operate in ‘internal’ mode (i.e. private IP).



The ‘Standard SKU’ load balancer adds support for zone redundancy.



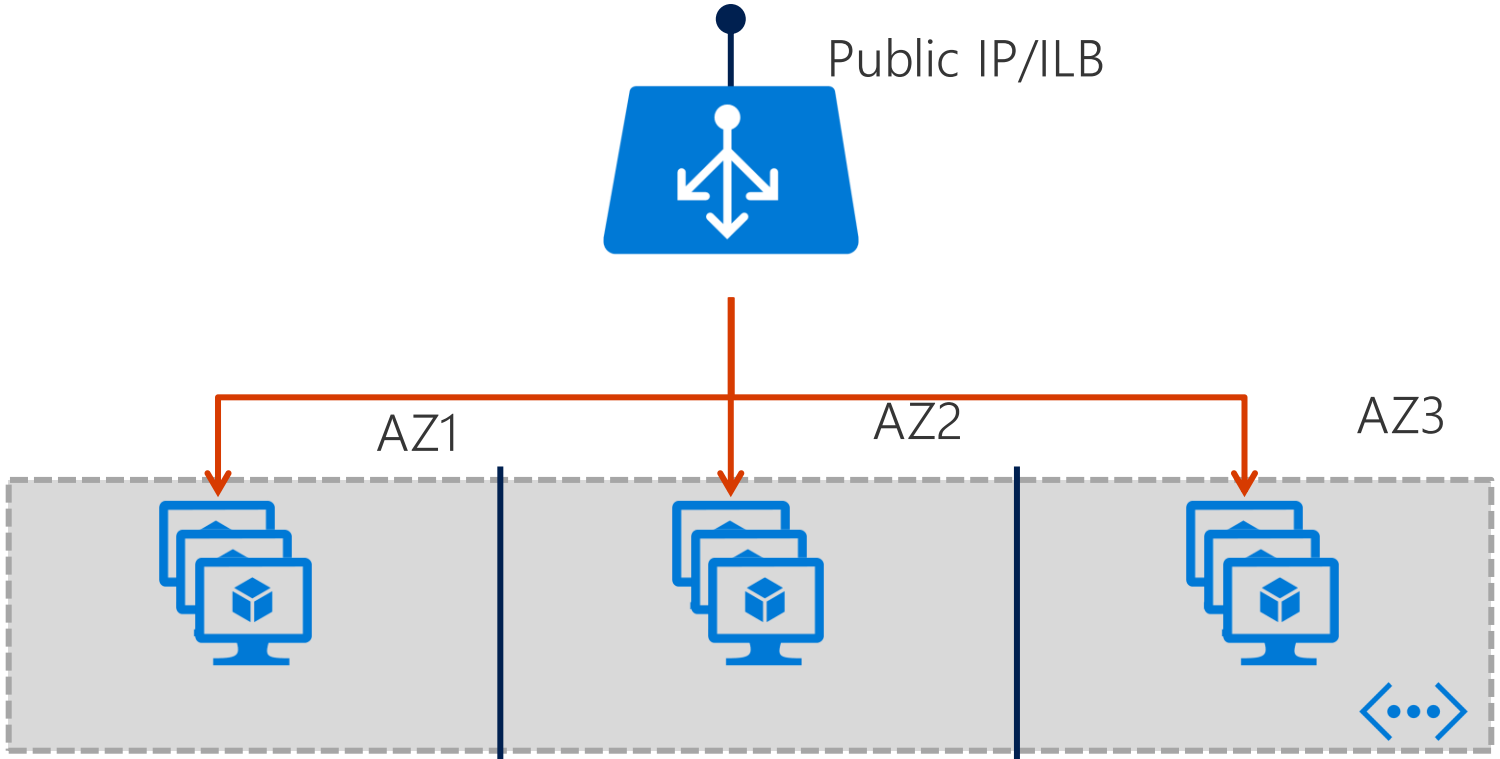
Azure Load Balancer (Standard)

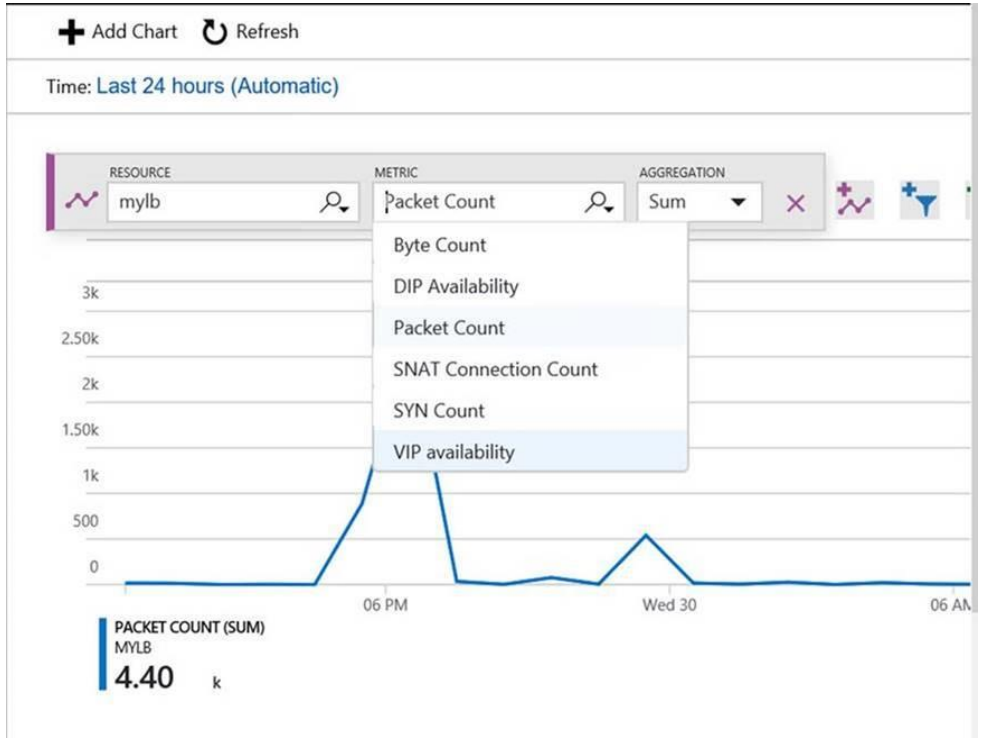
1000 backend VMs possible

High availability through regional anycast IPs – single IP across AZs

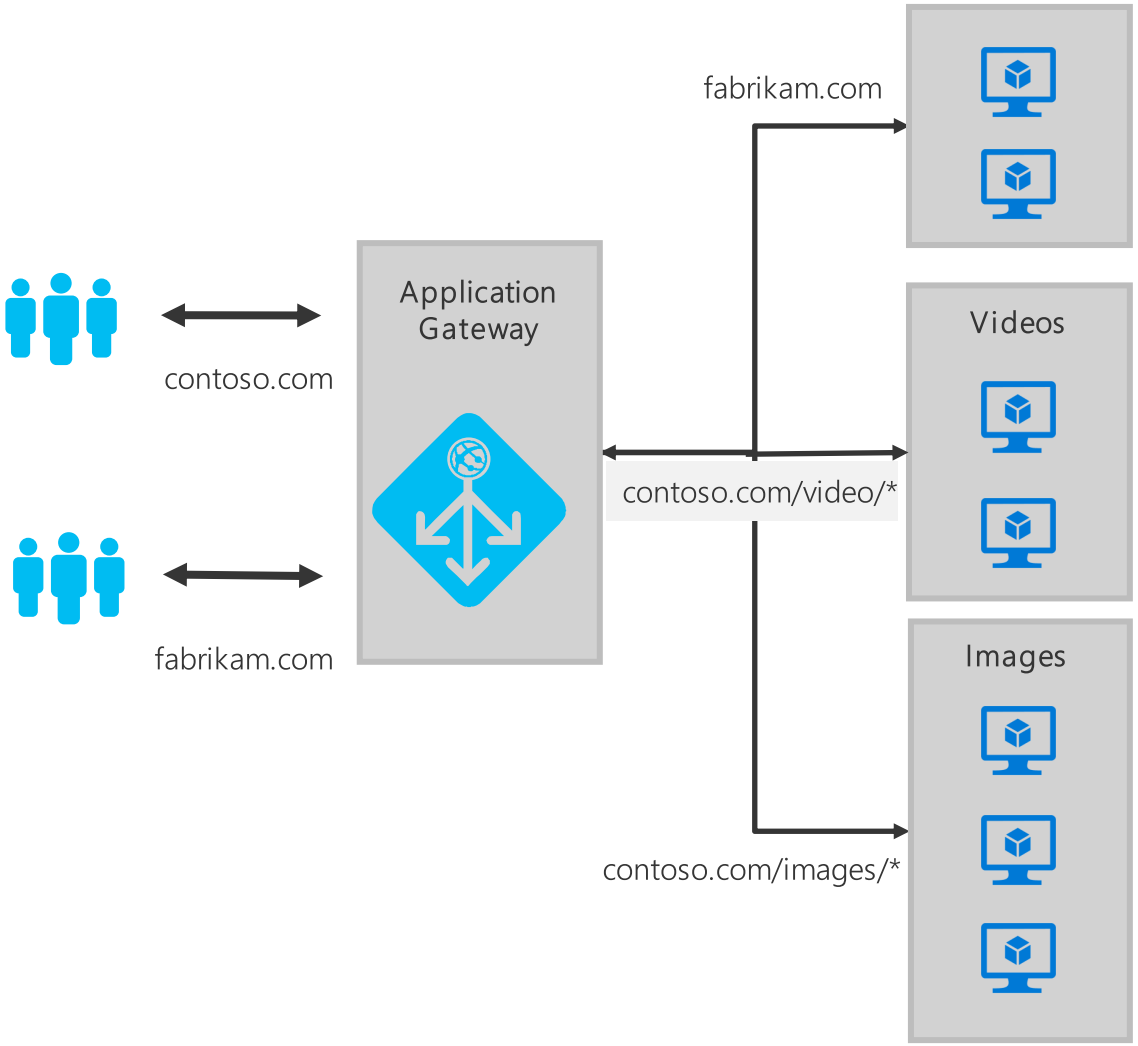
Drastically simplified NVA Resiliency – HA Ports

Extensive health and diagnostic metrics





App Gateway Multi-site and URL base routing



Layer 7 Routing

- HTTP round robin
- Cookie based session affinity
- Multi-site hosting
- URL based routing
- Support Web Apps
- Redirection

Security

- SSL termination
- SSL Policy (protocol and cipher)
- End to end SSL

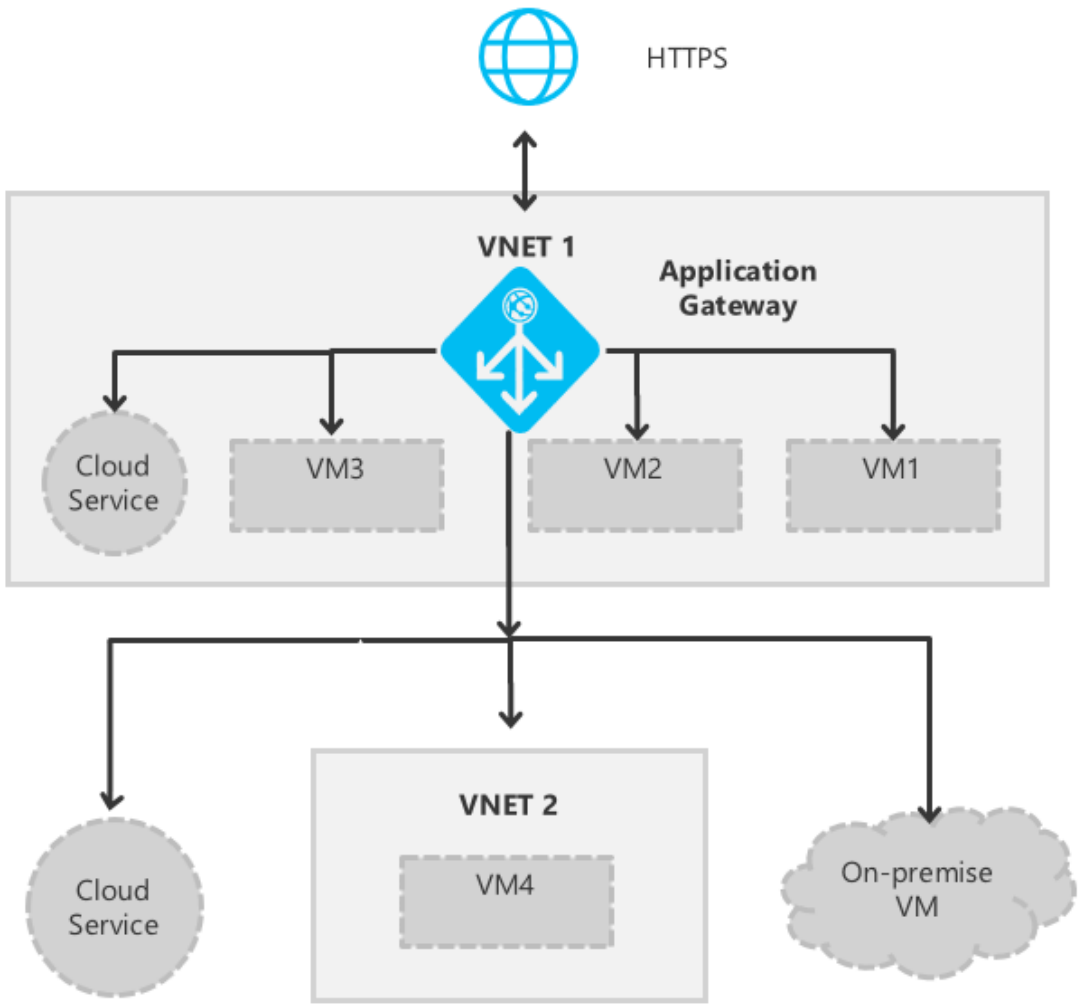
Application Gateway and WAF

Backend choices

- VMs via NICs
- Internal IP
- External Public IP
- VMSS
- Azure Web Apps
- Containers

Connectivity Options

- VMs in same VNet
- VMs across connected VNets
- Cloud services
- Hybrid connectivity to on premises VMs
- External servers



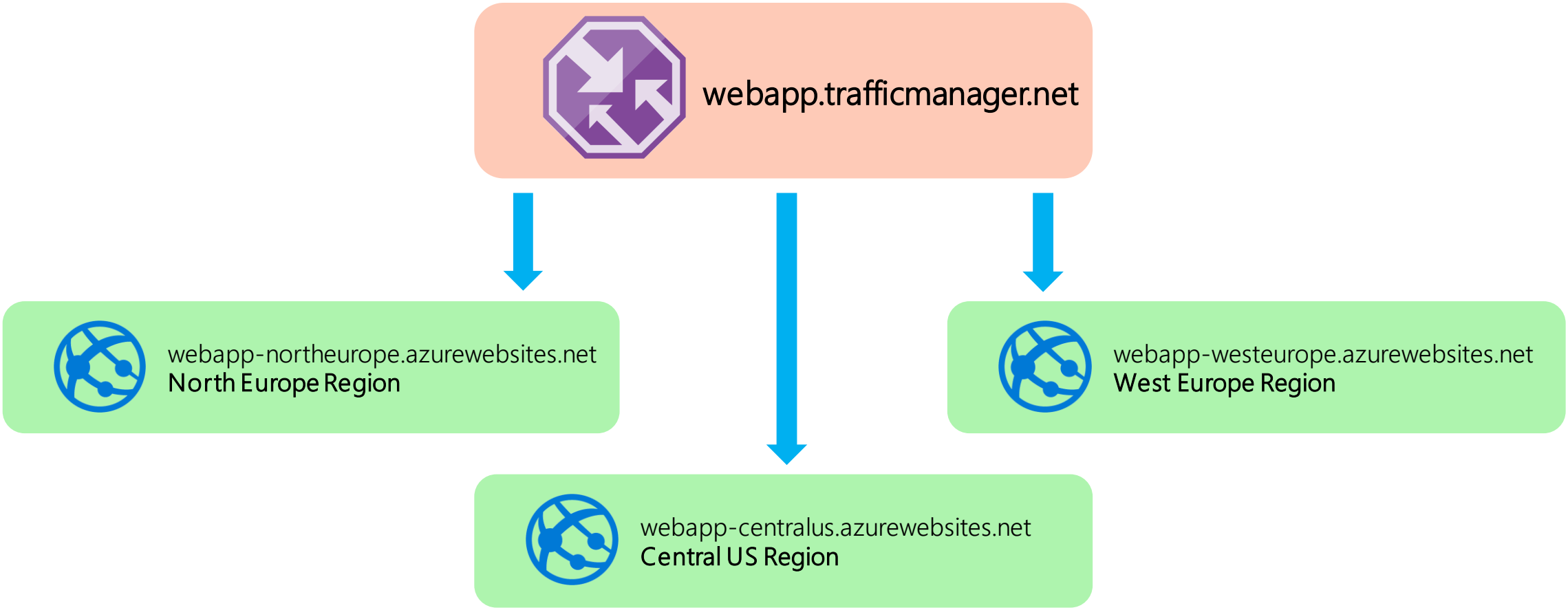
Layer 7 load balancing

- SSL termination and policy
- HTTP, HTTPS, HTTP/2, WebSocket
- URL path based routing
- Host header/SNI based routing
- Cookie based session affinity
- Redirection support at Gateway
- Support multi-tenant backend—Azure Web Apps

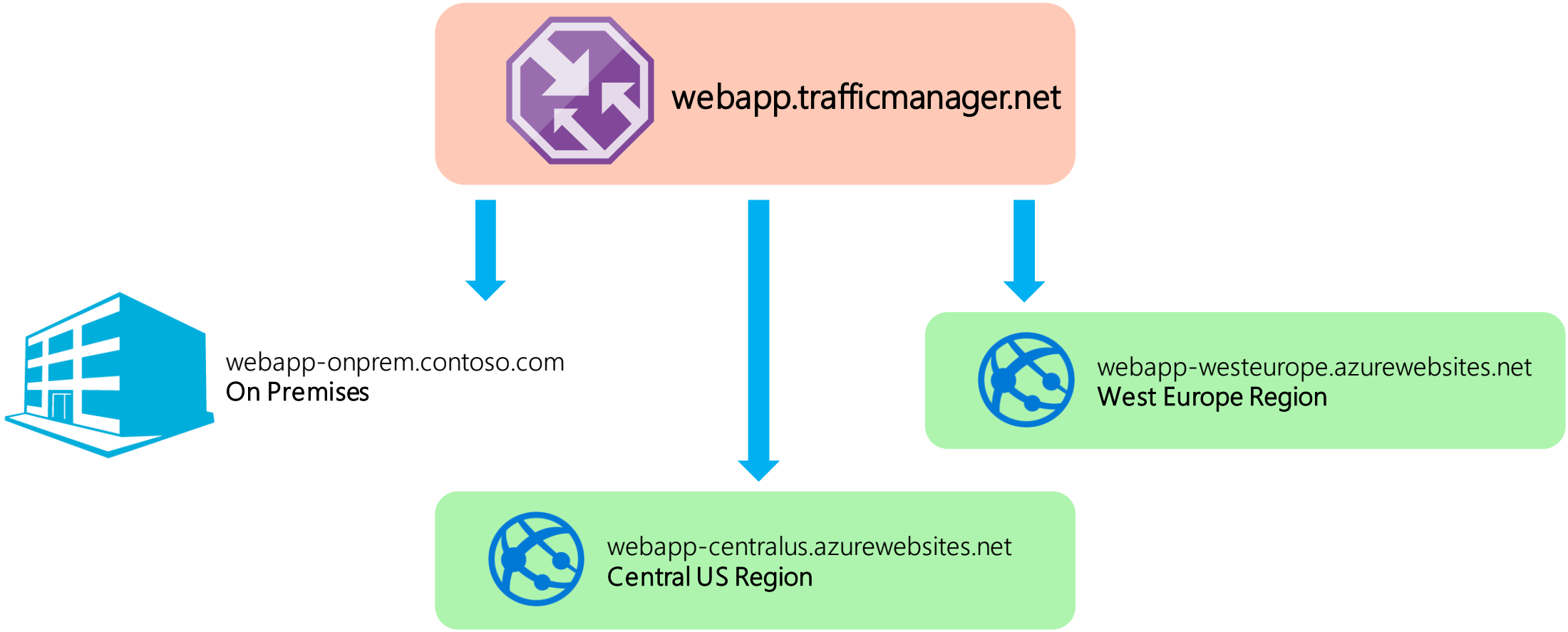
Web Application Firewall

- Support OWASP ModSecurity CRS 3.0
- Enable/Disable Rules or Rule Groups
- WAF logs Integrated with Azure Monitor
- WAF integrated with Azure Security Center

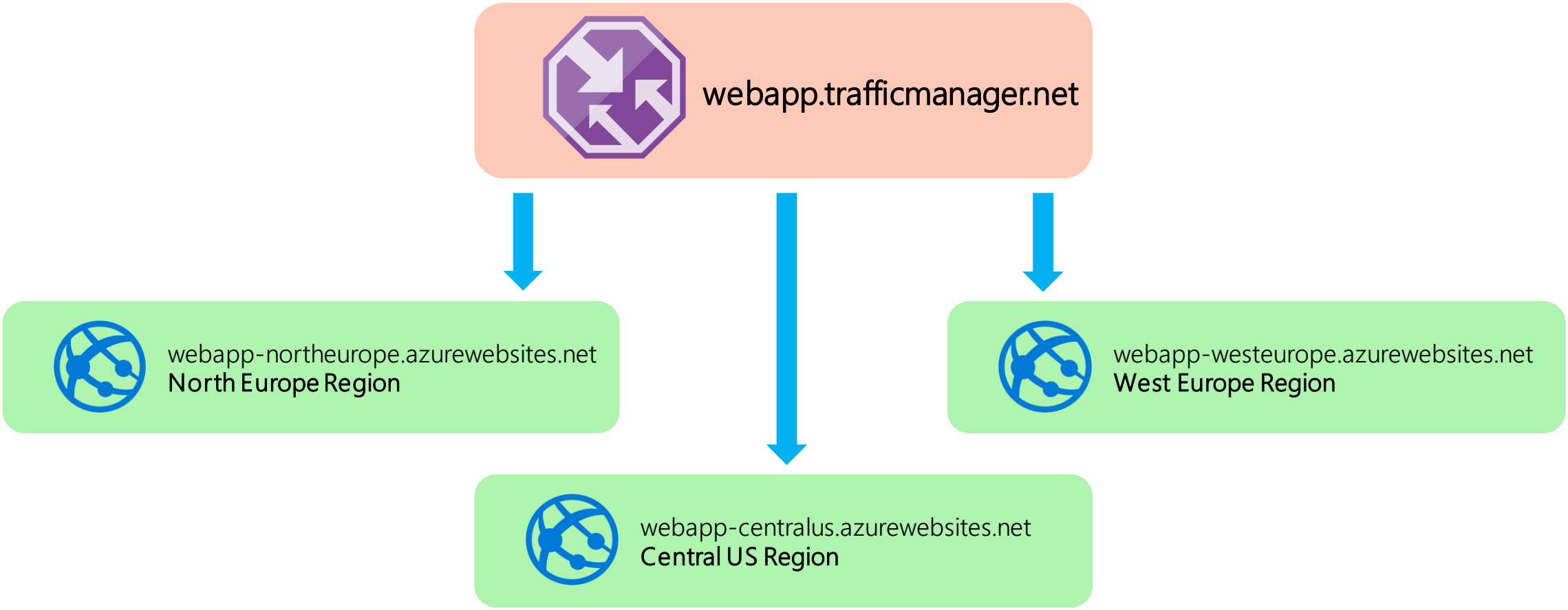
DNS based global load balancing with Traffic Manager



Traffic manager can also work with non-Azure based end points.



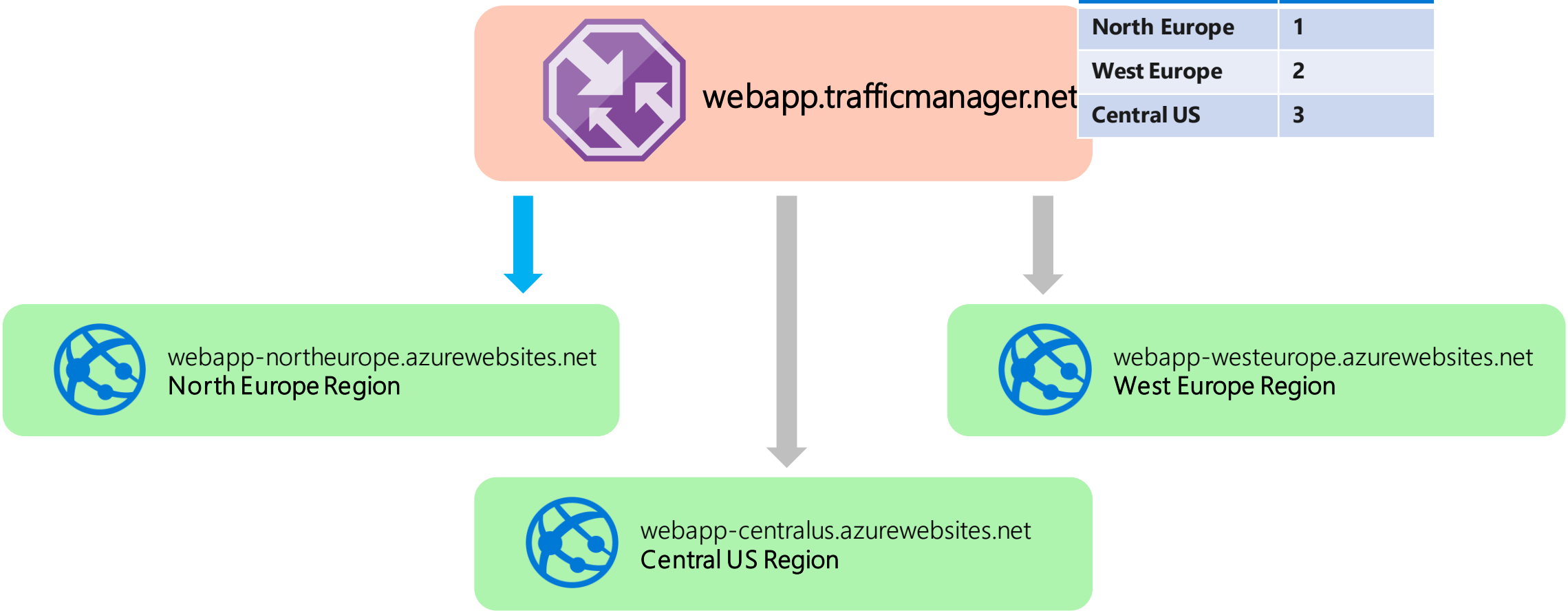
Traffic Manager has four routing methods.



1. Priority Routing

Use a primary endpoint for traffic; failover when primary is unavailable.

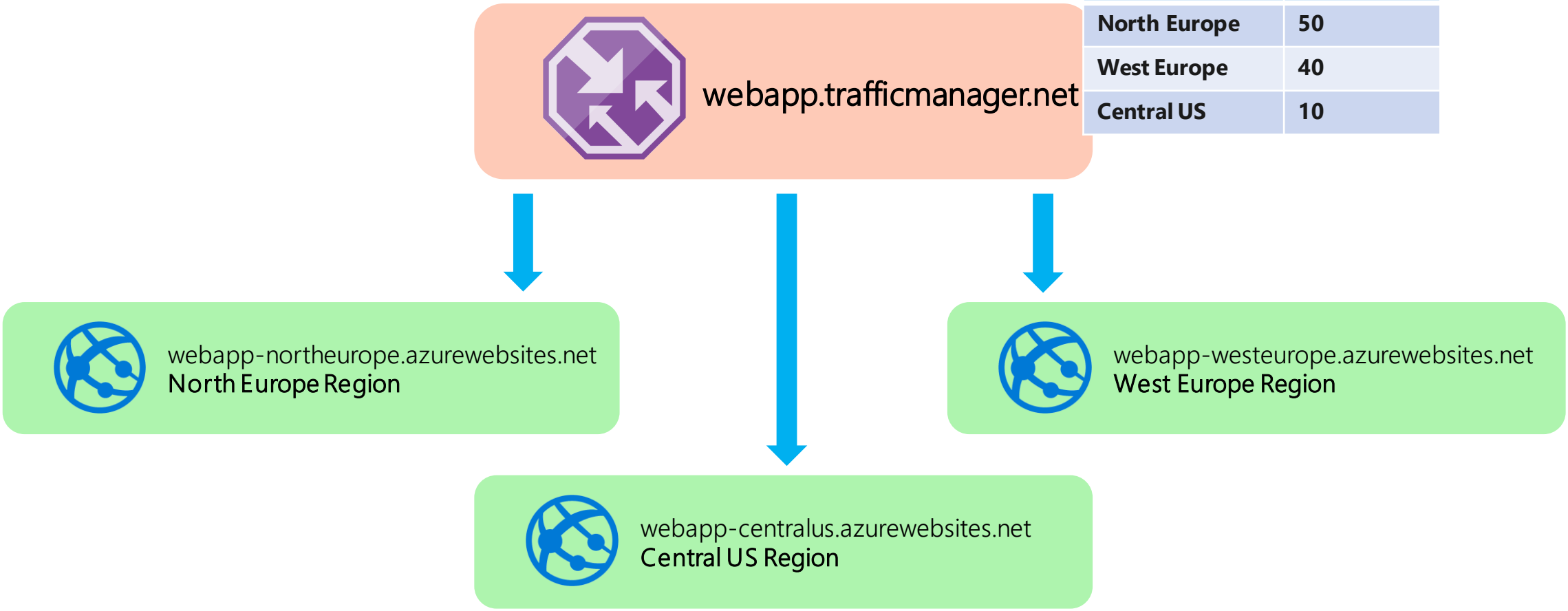
End Point	Priority
North Europe	1
West Europe	2
Central US	3



2. Weighted Routing

Distribute traffic according to weights.

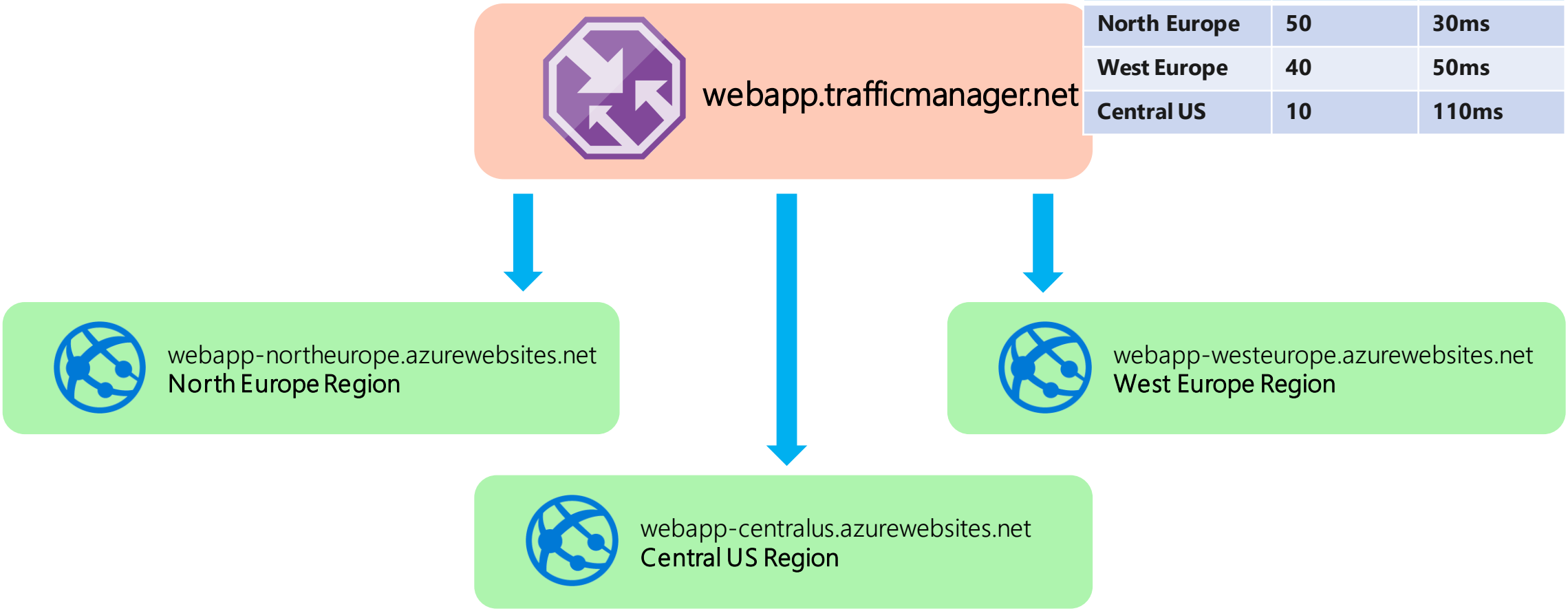
End Point	Priority
North Europe	50
West Europe	40
Central US	10



3. Performance Routing

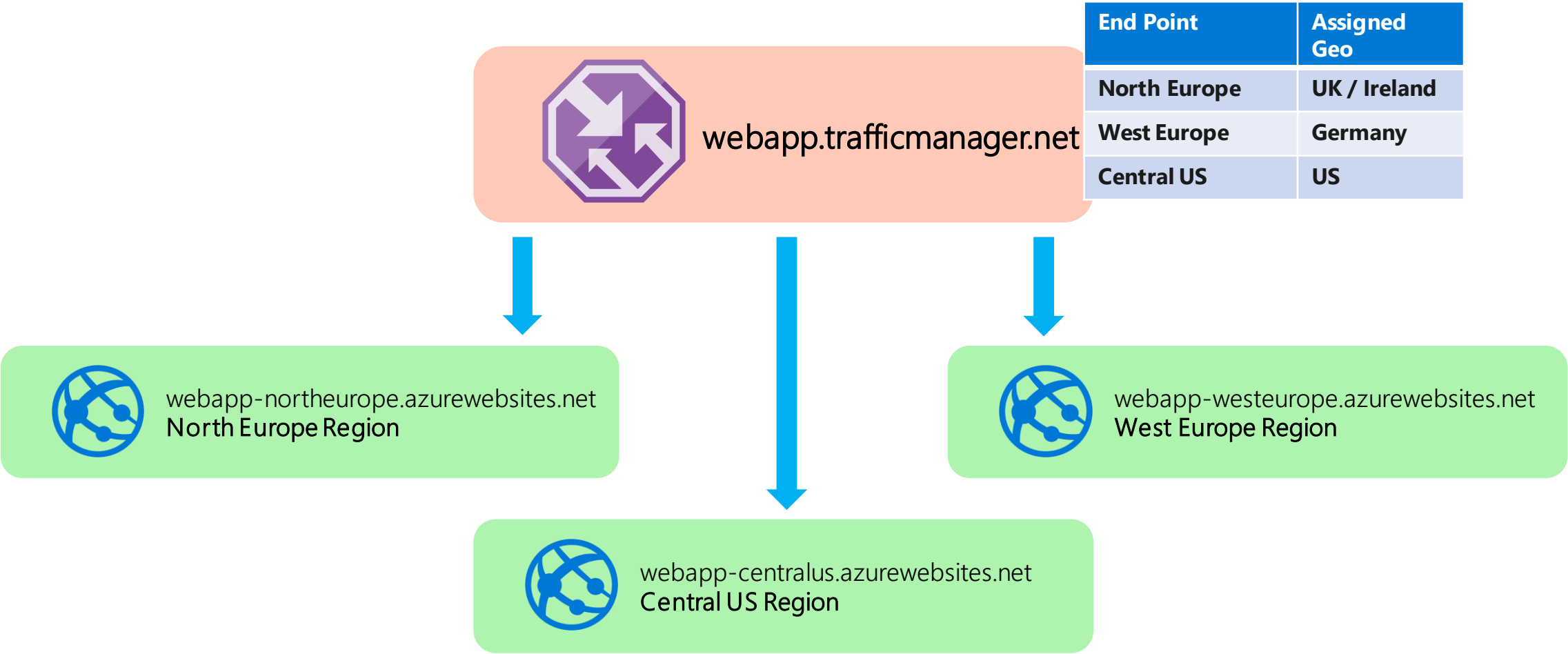
Direct users to the 'closest' end point, based on latency.

End Point	Priority	Latency
North Europe	50	30ms
West Europe	40	50ms
Central US	10	110ms



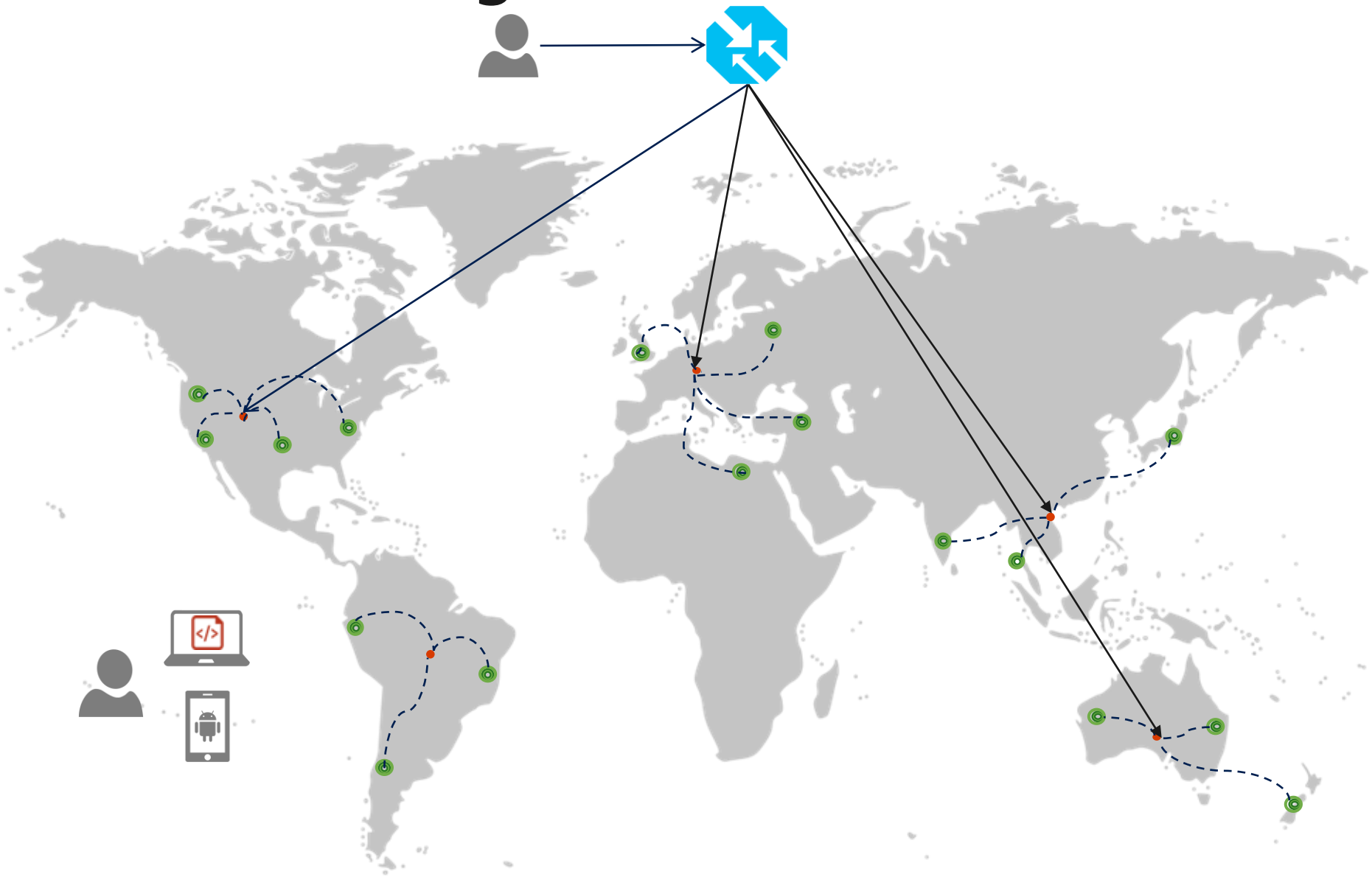
4. Geographic Routing

Direct users based on the geographic location the DNS query came from.



End Point	Assigned Geo
North Europe	UK / Ireland
West Europe	Germany
Central US	US

Traffic Manager Real User Measurements



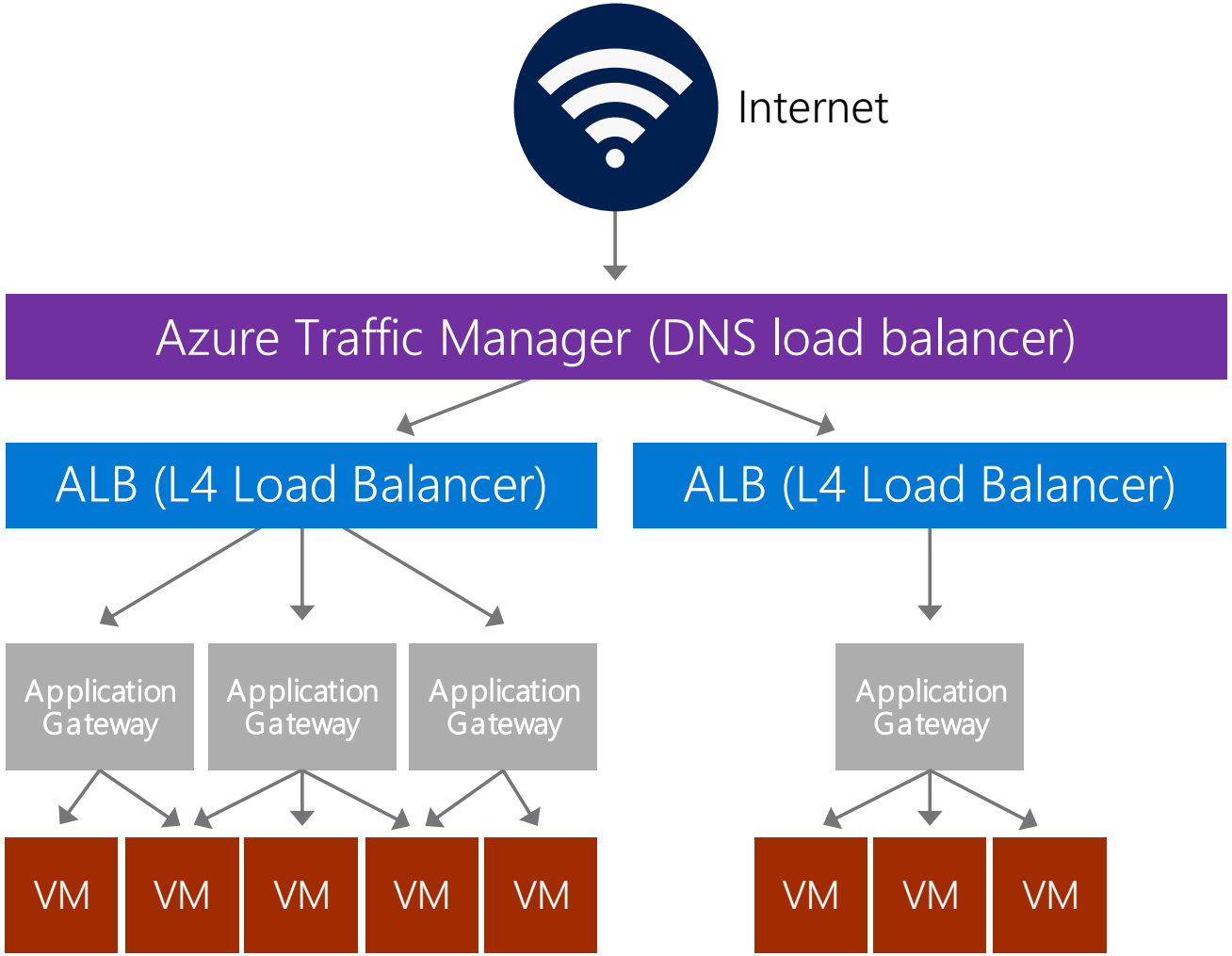
Increased accuracy in routing decisions by leveraging Real User Measurements (RUM) taken from where your end users connect from

Latency metrics are obtained from your customer base ensuring that all end users networks relevant to you are covered

Integration with Visual Studio Mobile Center

Azure Load Balancer Hierarchy

AZURE SERVICE	WHAT	EXAMPLE
Traffic Manager (TM)	Cross-region redirection and availability	http://news.com ➔ apac.news.com ➔ emea.news.com ➔ us.news.com
Azure Load Balancer (ALB, ILB)	In-region scalability and availability	emea.news.com ➔ AppGw1 ➔ AppGw2 ➔ AppGw2
Azure Application Gateway (AppGW)	URL/content-based routing and Load Balancing	news.com/top news news.com/sports news.com/images
VMs	Web servers	IIS, Apache, Tomcat



Symantec Use Cases

Norton and Traffic Manager

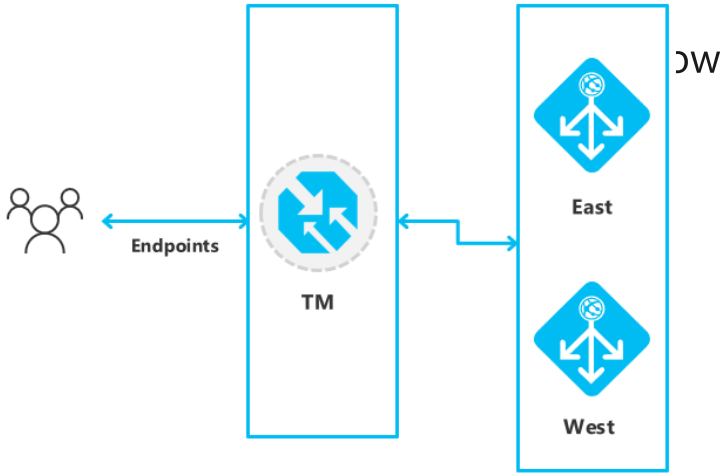
Service

- All Norton services
- Including internal communication

Scale

- Routing 150+ million endpoints (clients)
- 1+ billion Queries / Month

Reliability & Performance



SPOC and Azure Load Balancer

Service

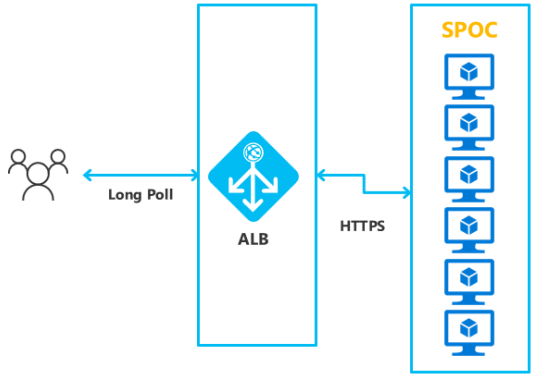
- Single Point of Contact, endpoint notification
- Multi Region

Scale

- 39k requests / sec, 2+ billion req / day
- 10-20 million concurrent connections
- Average time to service request < 2ms

Azure

- Traffic Manager Balance 50/50 (normal)
- Eliminated Public IP needs
- 4-30 minute Long Polling
- 60 ms latency reduction (world wide)



Shasta and Application Gateway

Service

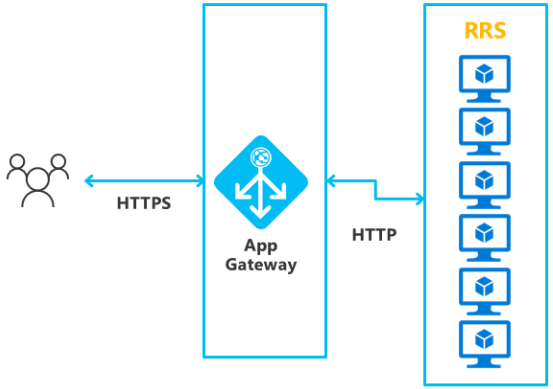
- File Reputation
- Protection for Norton & Symantec Endpoints
- Multi Region

Scale

- 25k requests / sec, 1.5+ billion req / day
- 2ms Service -> App GW Response Latency
- Maximum worldwide response < 5s

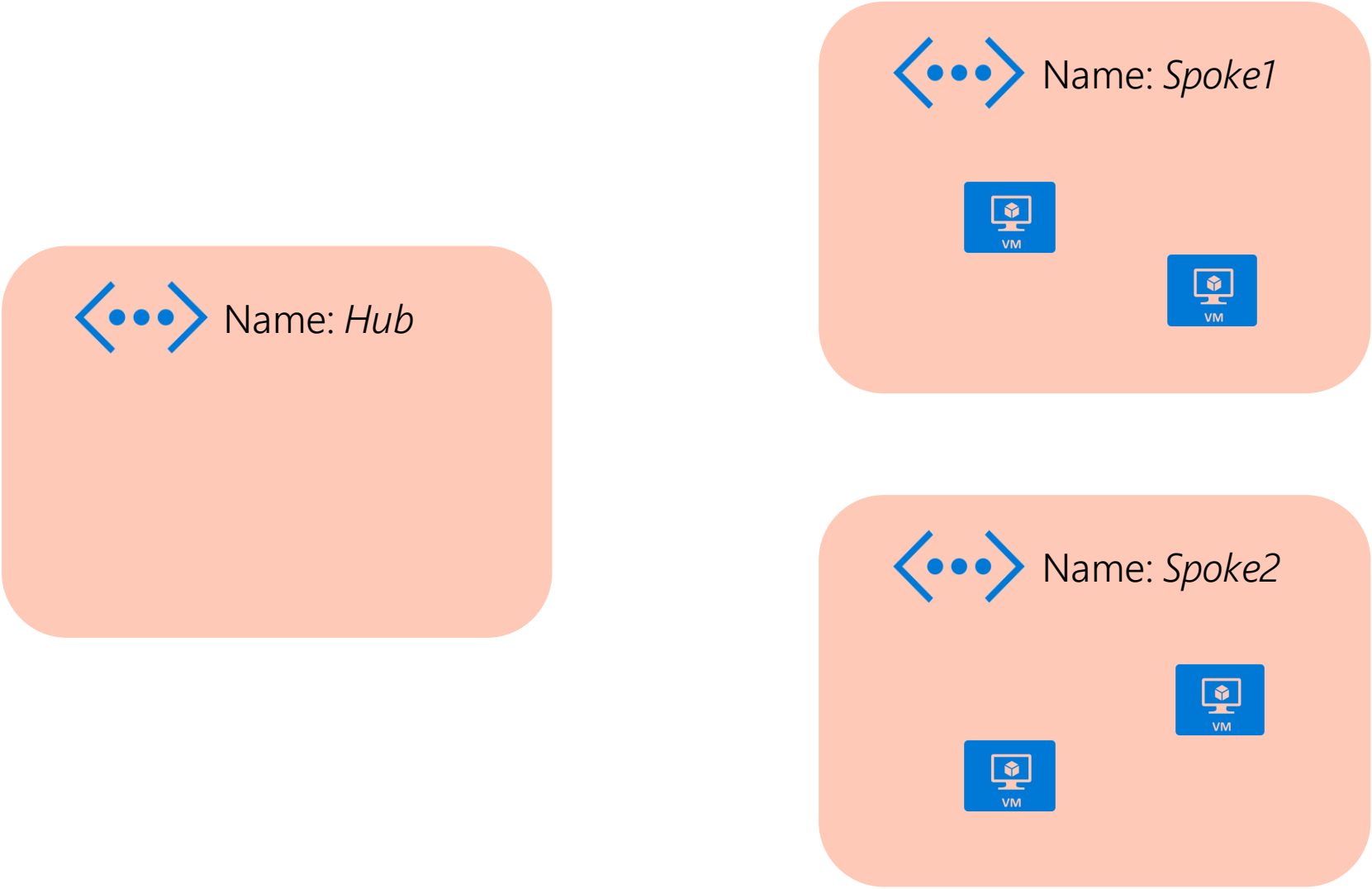
Azure

- Leverage Application GW
  - Hardware in DC's vs Managed Service
  - Virtual Appliances

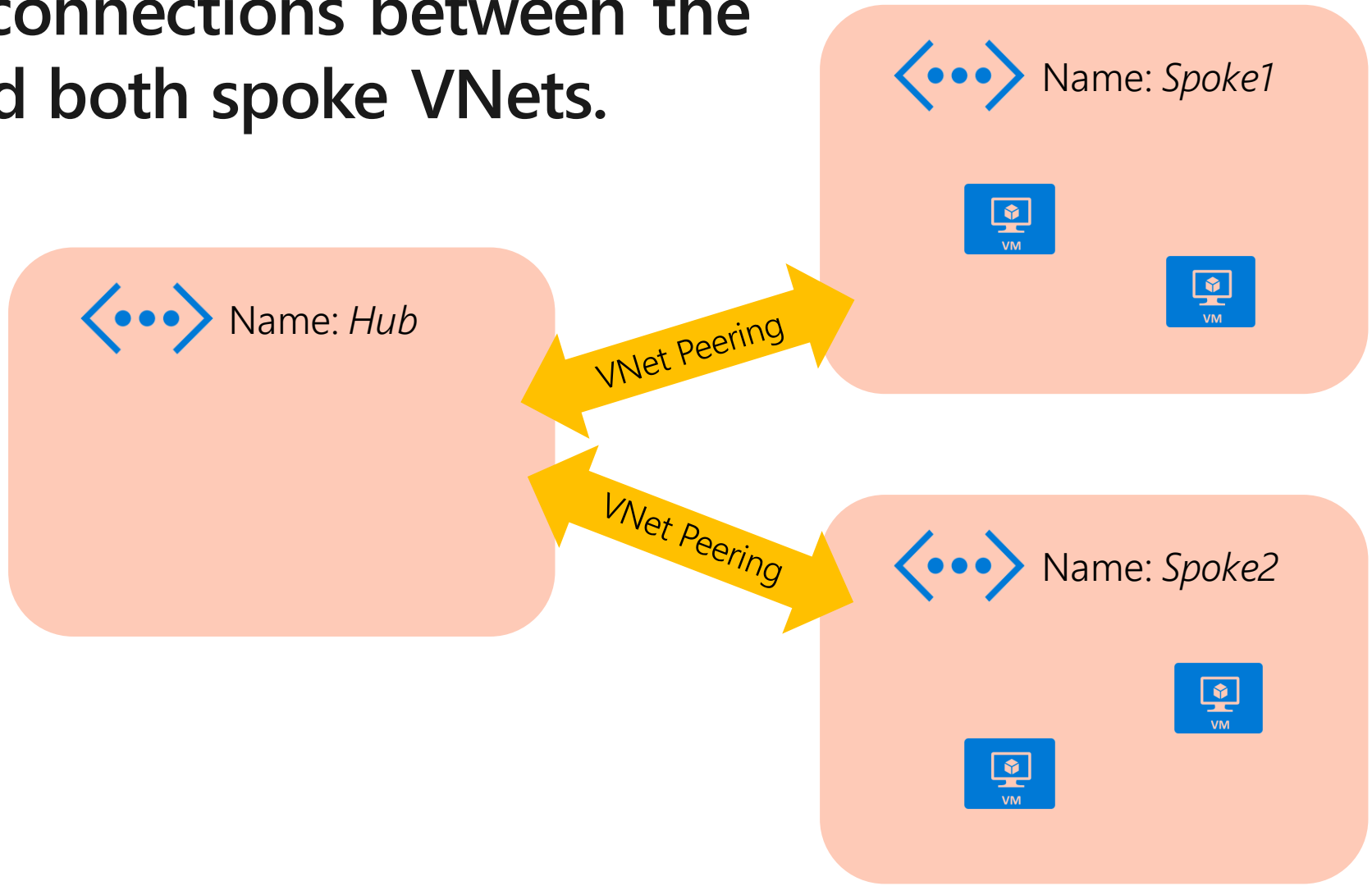


Azure Network Topologies

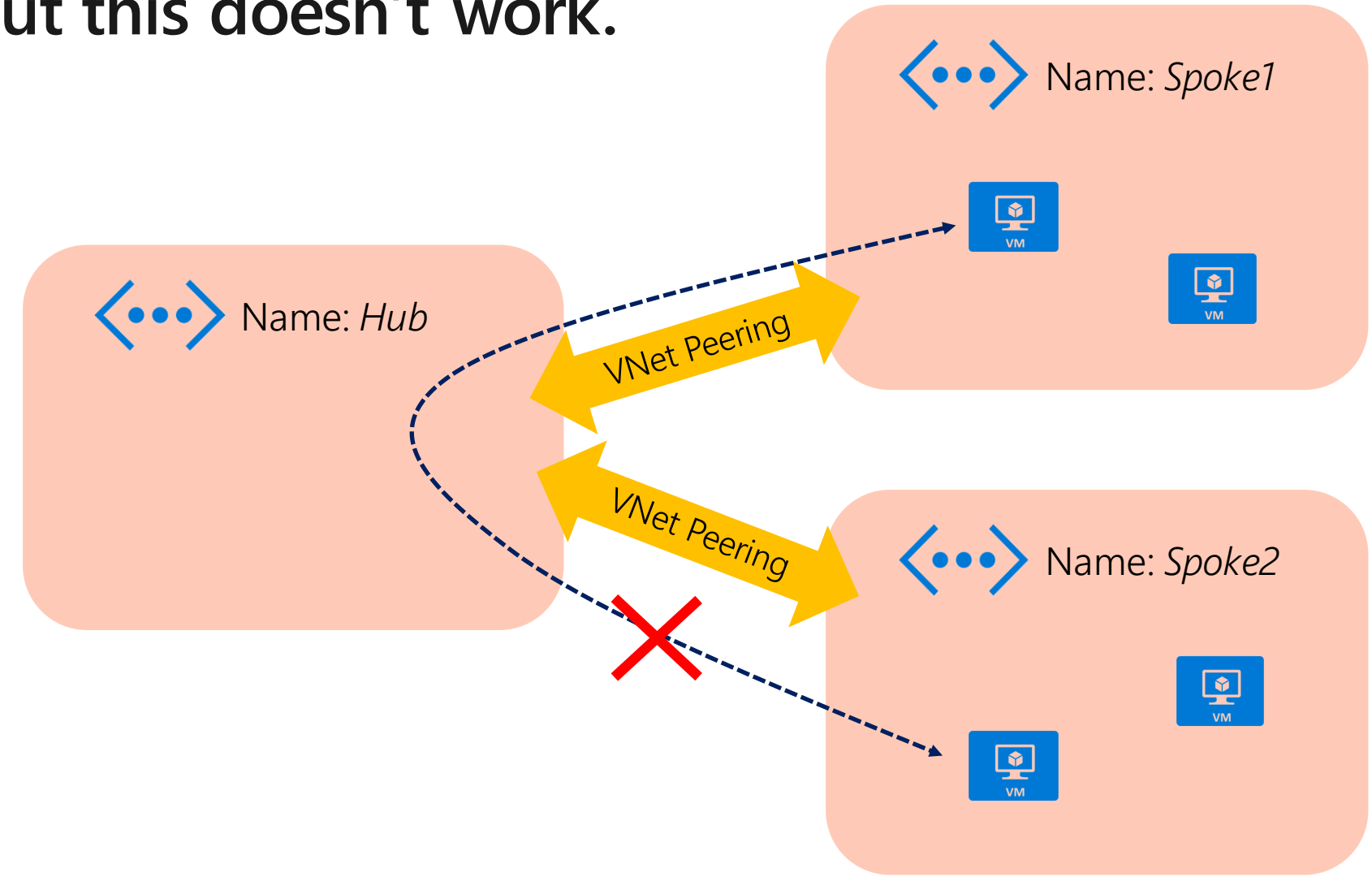
How do we build multiple VNet topologies in Azure?



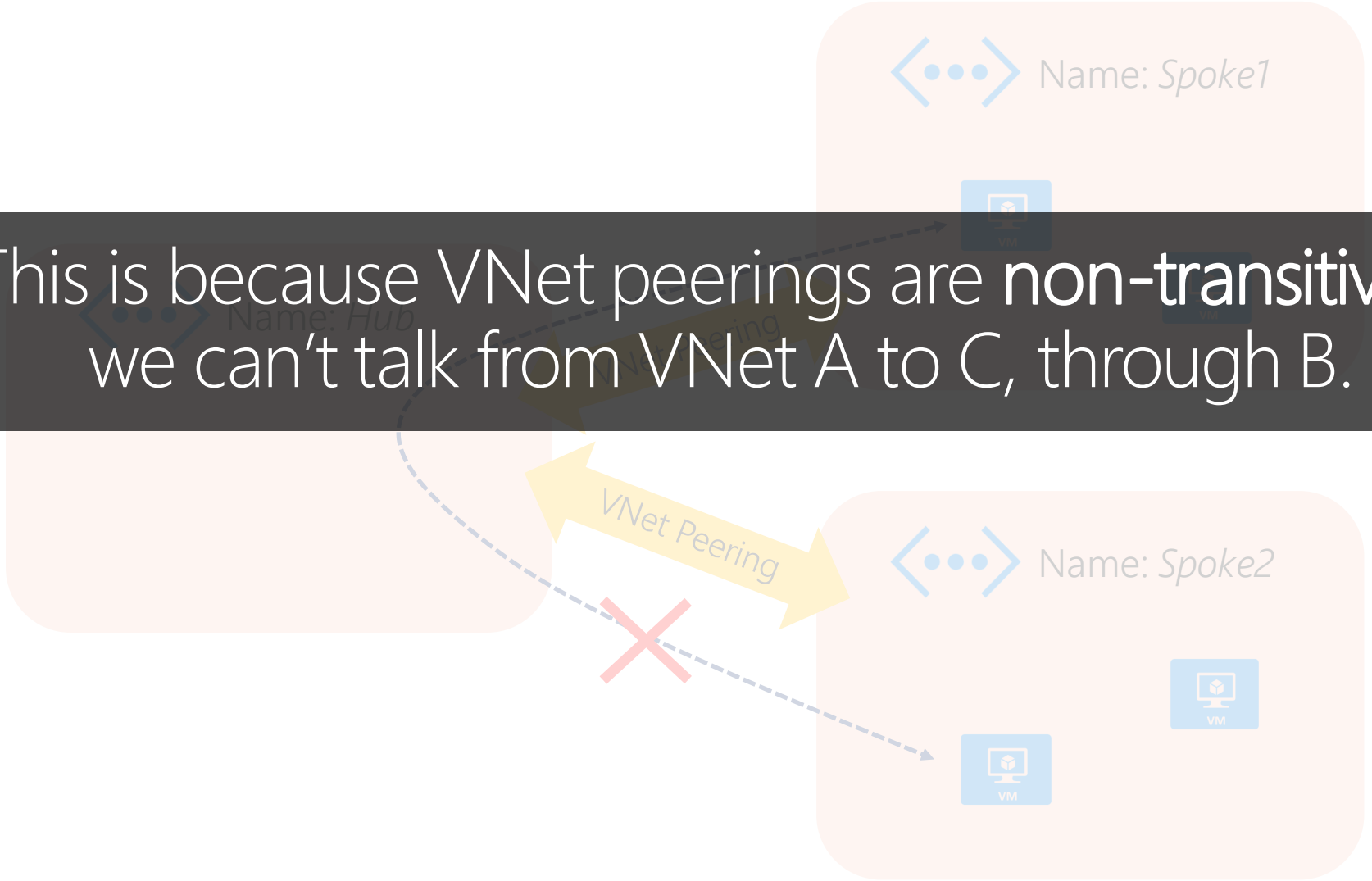
Let's say we use VNet peering to create connections between the hub and both spoke VNets.



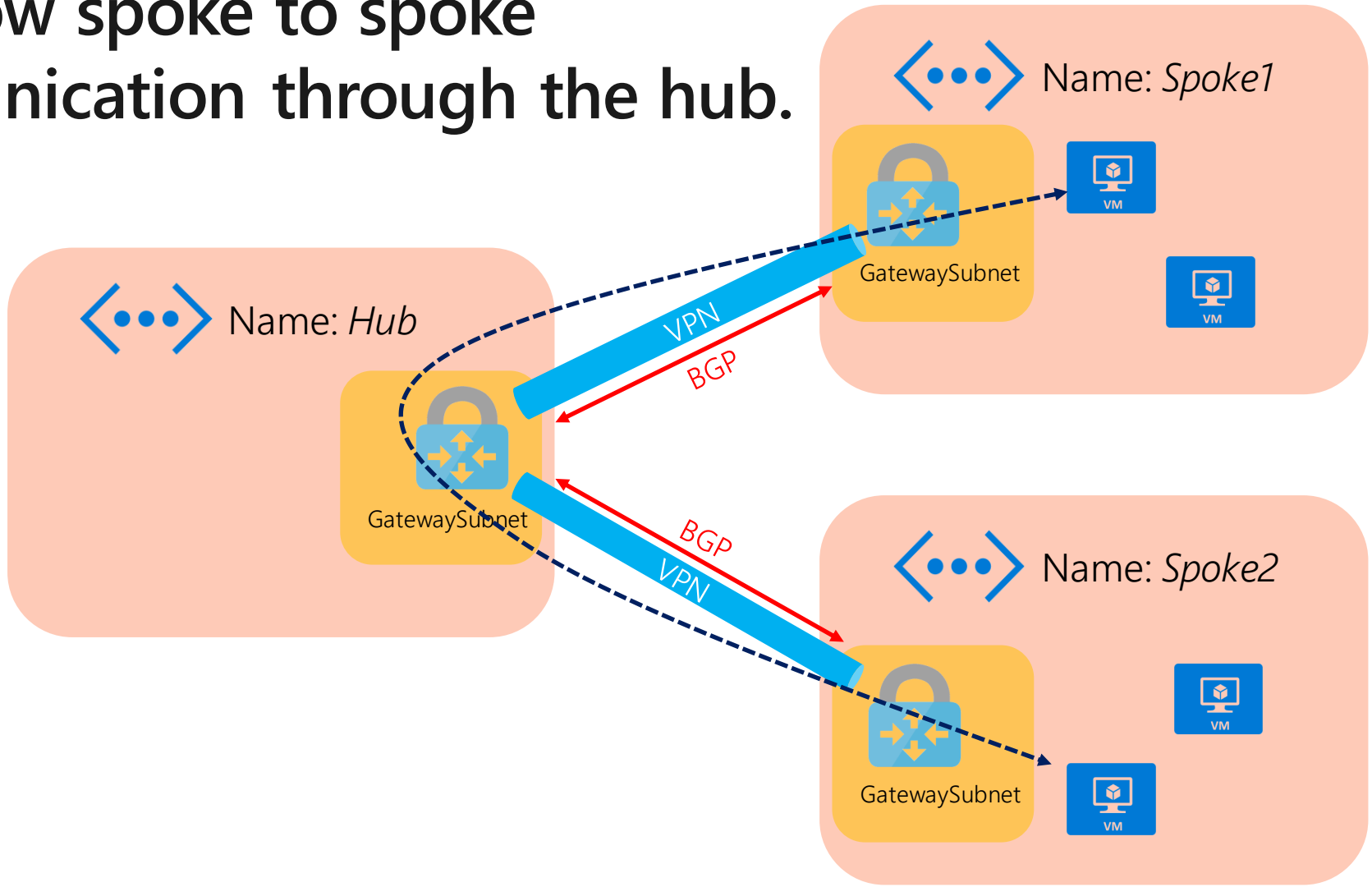
We might expect Spoke1 VMs to be able to talk to Spoke2 VMs, but this doesn't work.



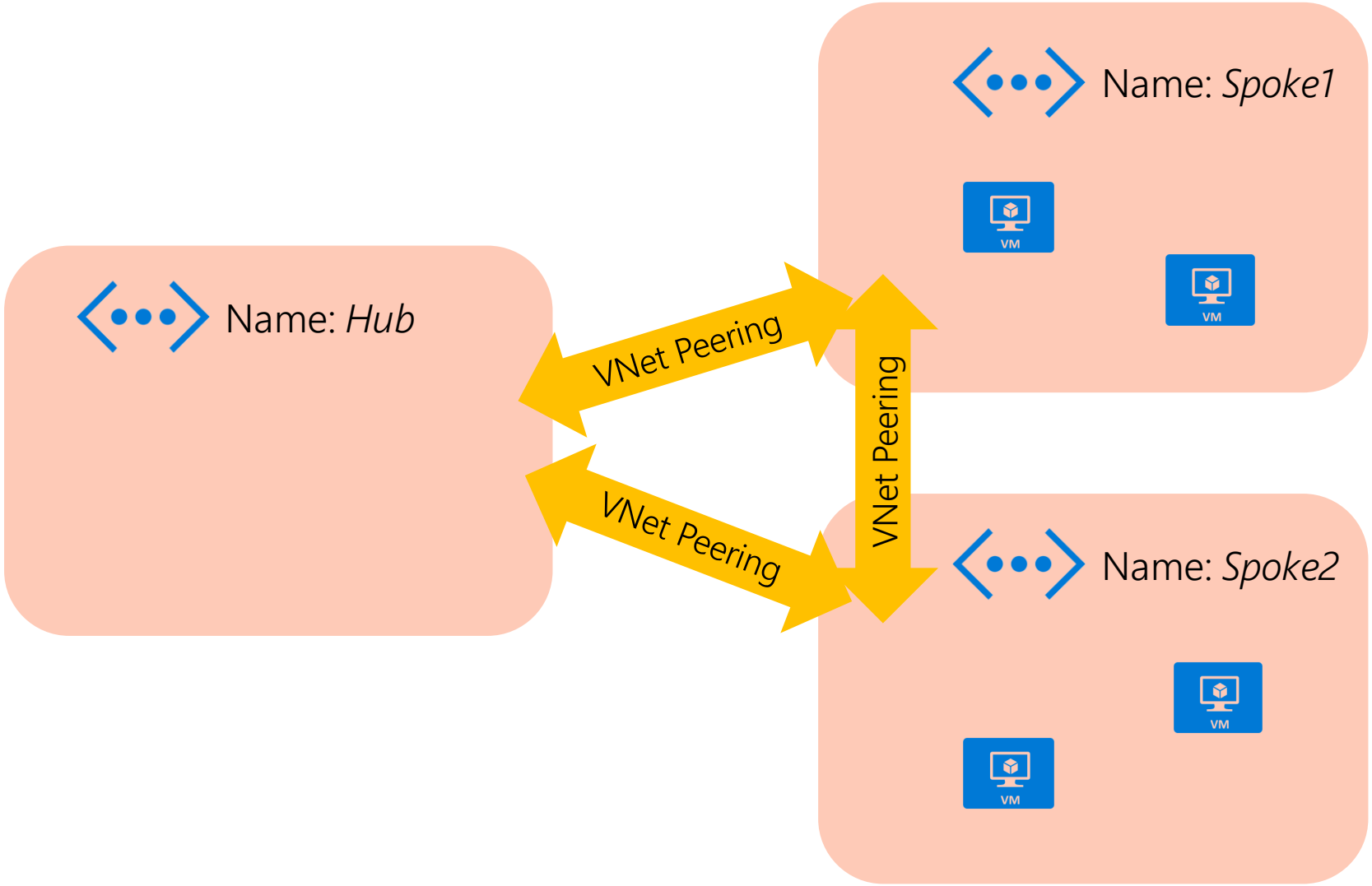
This is because VNet peerings are **non-transitive** – we can't talk from VNet A to C, through B.



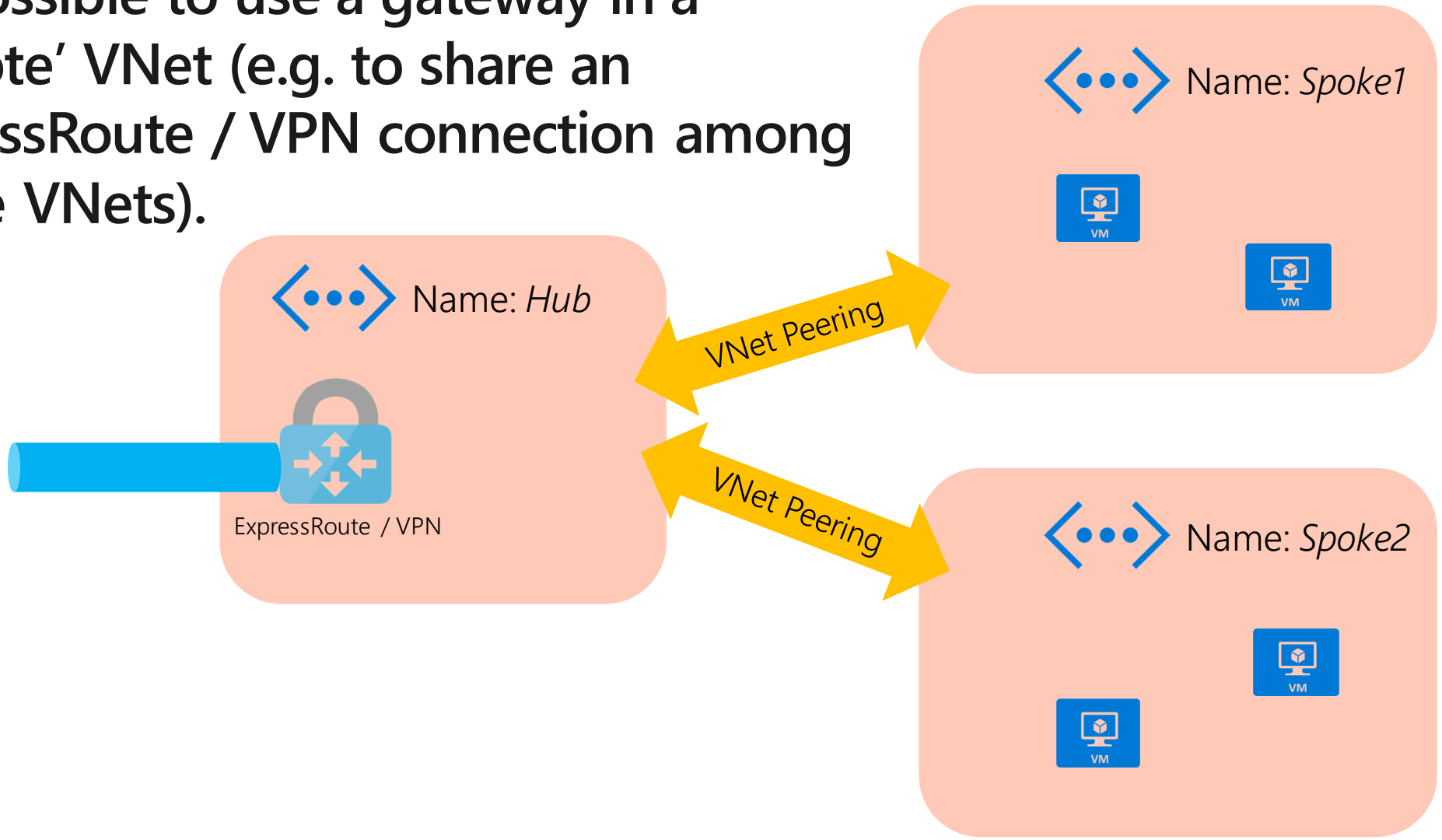
Using a site-2-site VPN with BGP will allow spoke to spoke communication through the hub.



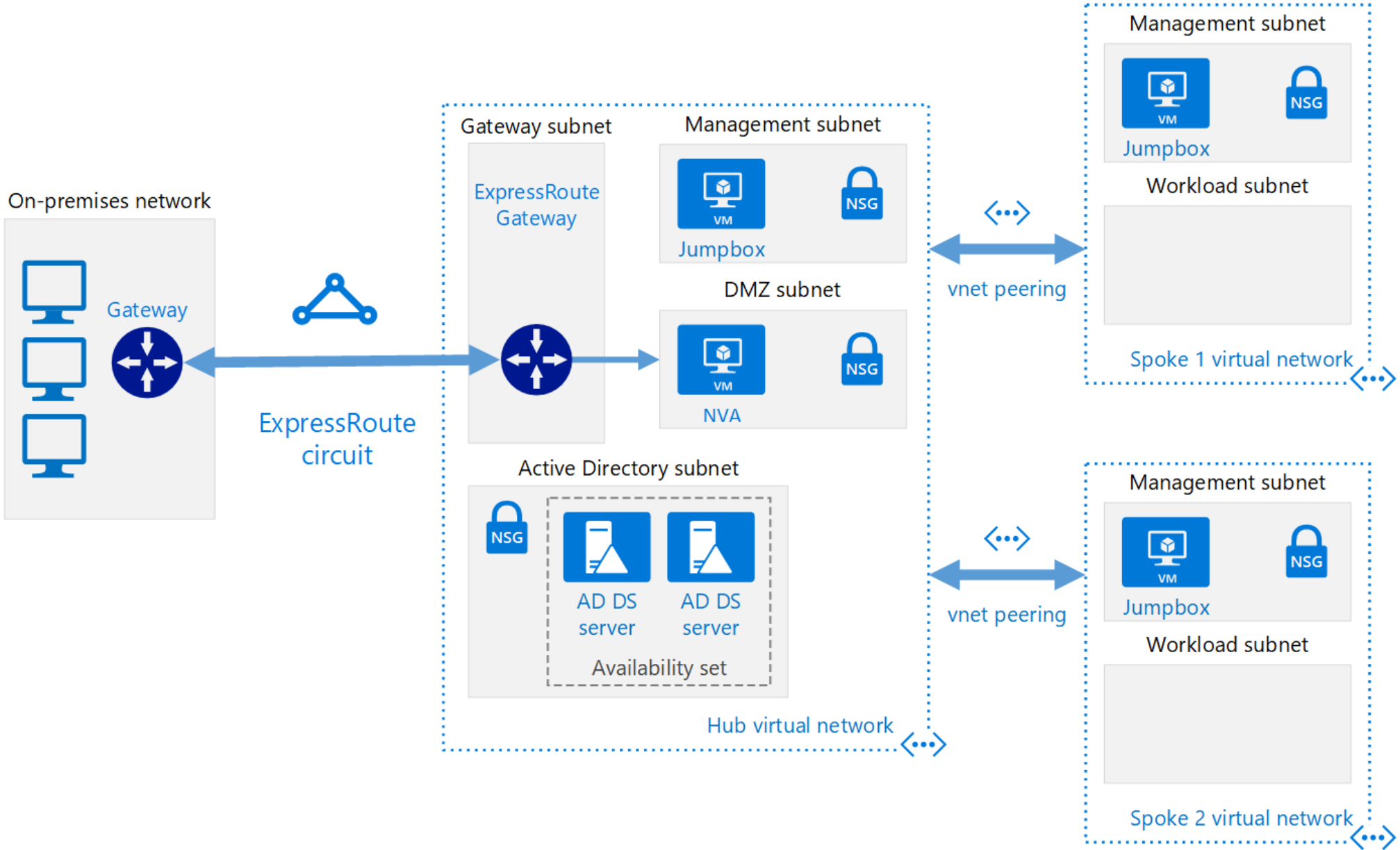
Alternatively, a full mesh of VNet peerings will work.



It's possible to use a gateway in a 'remote' VNet (e.g. to share an ExpressRoute / VPN connection among spoke VNets).



Example of HUB and Spoke



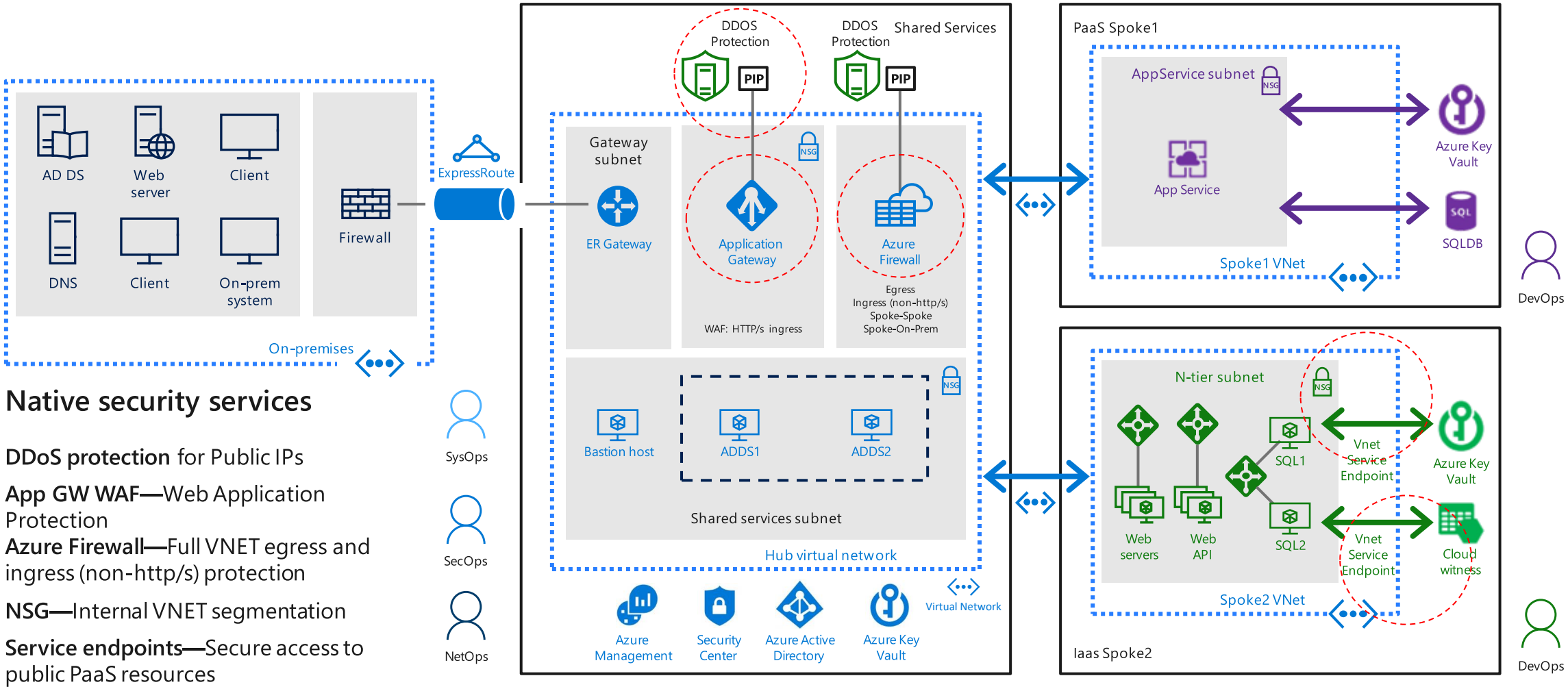
Each spoke/Vnet can be a LoB in his own subs.

Granular security constrains by Vnet & subnets

Isolation by tenant but still applying global controls

Forced tunneling

Hub & spoke architecture native security services



Native security services

DDoS protection for Public IPs

App GW WAF—Web Application Protection

Azure Firewall—Full VNET egress and ingress (non-http/s) protection

NSG—Internal VNET segmentation

Service endpoints—Secure access to public PaaS resources

Azure Firewall

Azure Firewall

Cloud native stateful Firewall as a service

A first among public cloud providers

Central governance of all traffic flows

- Built-in high availability and auto scale
- Network and application traffic filtering
- Centralized policy across VNets and subscriptions

Complete VNET protection

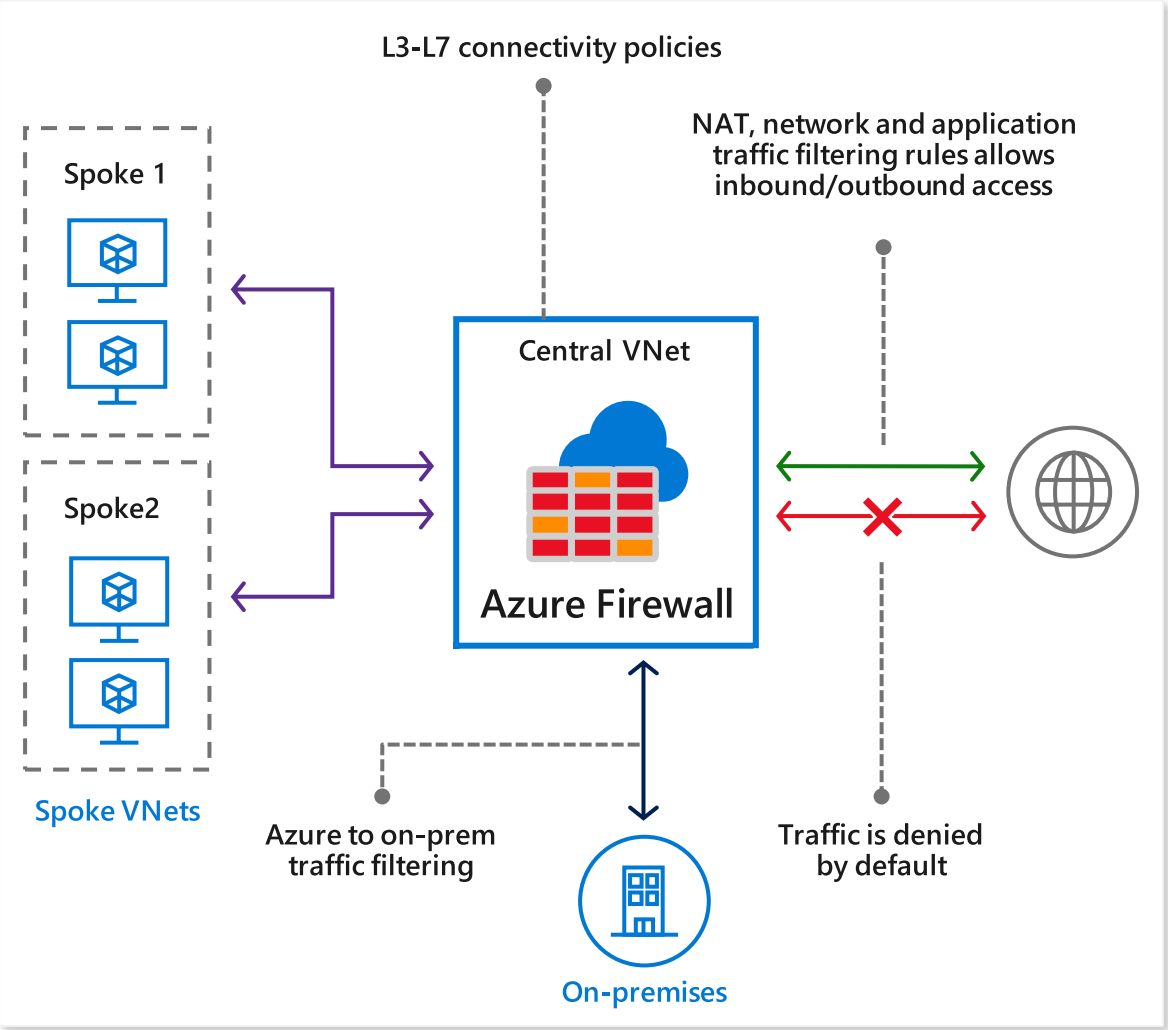
- Filter Outbound, Inbound, Spoke-Spoke and Hybrid Connections traffic (VPN and ExpressRoute)

Centralized logging

- Archive logs to a storage account, stream events to your Event Hub, or send them to Log Analytics or Security Integration and Event Management (SIEM) system of choice

Best for Azure

- DevOps integration, FQDN Tags, Service Tags, Integration with ASE, Backup and other Azure services



Azure Firewall

Application rules

- FQDN Filtering
- FQDN Tags (e.g., Azure Backup, App Service Environment)
- Default infrastructure rule collection

Fully stateful network rules

NAT support

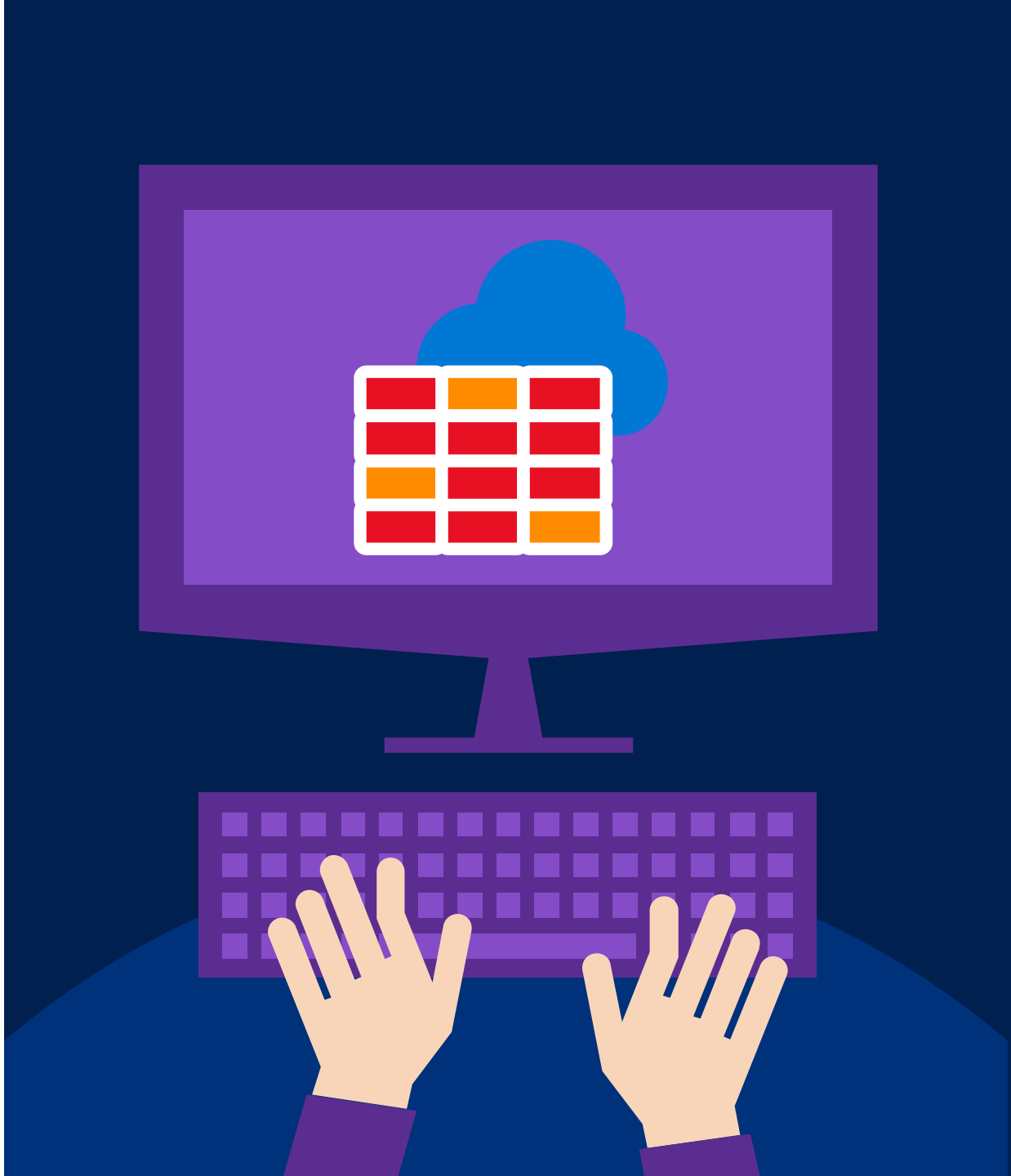
- Default Source Network Address Translation (SNAT)
- Destination Network Address Translation (DNAT)

Monitoring

- Azure monitor logging
- Azure monitor metrics

Support for inbound and hybrid connections

Network watcher integration



Inbound traffic filtering

Application gateway WAF is the preferred service for inbound application level HTTP/S protection

Use Azure Firewall inbound network level protection for:

Non-HTTP/S protocols (e.g., SSH, RDP, FTP)

HTTP/S network level protection along with SaaS based WAF provider

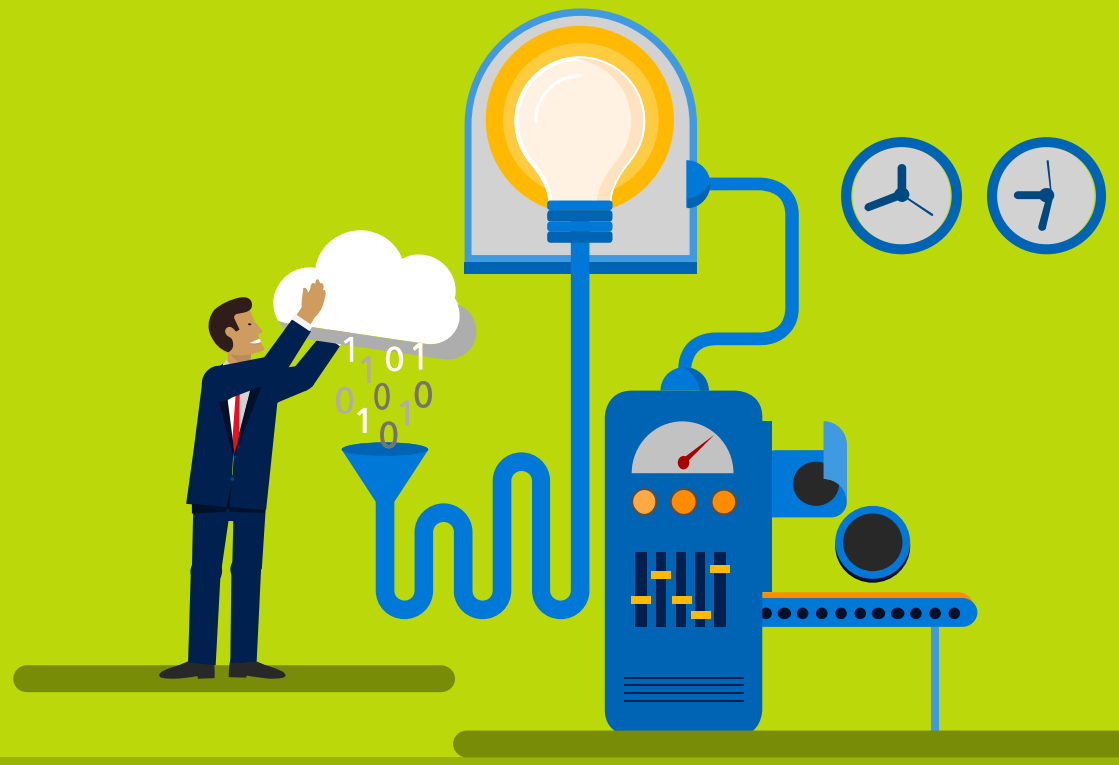
Destination Network Address Translation (DNAT)

Inbound traffic filtering is enabled by mapping of your firewall public IP and port to a private IP and port

Inbound rules processing logic

DNAT rules implicitly add a corresponding network rule to allow the translated traffic. You can override this behavior by explicitly adding a network rule collection with deny rules that match the translated traffic

Application rules are not applied on inbound translated traffic—this is on our roadmap



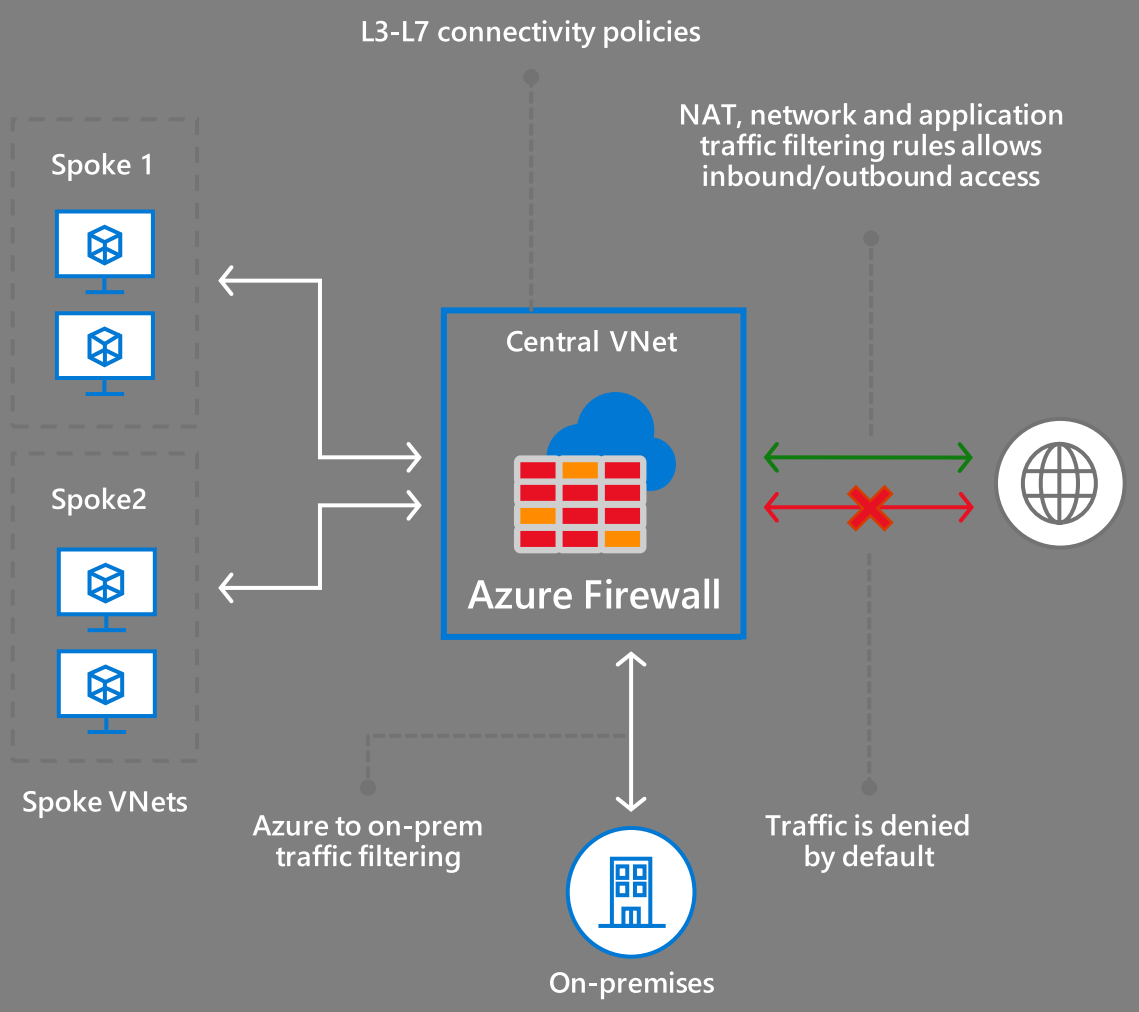
Azure Firewall for hybrid links

Traffic filtering between Azure VNETs and on-premises networks

Works with either Azure VPN Gateway or Express Route Gateway

No support for traffic routing from on-premises to internet*

*This is a key roadmap feature for Azure Firewall in a virtual WAN hub



Hybrid links filtering – Recommended Configuration

1. UDR on the spoke subnet pointing to Azure Firewall private IP as default gateway
   - BGP route propagation must be disabled on this route table
2. UDR on the hub gateway subnet pointing to Azure Firewall as next hop to spoke networks
   - Pointing to Azure Firewall as the default gateway is not supported on gateway subnets
3. No UDR on Azure Firewall subnet (it learns routes from BGP)
4. Allow spokes to use VPN/ER gateway in the hub
   - Set AllowGatewayTransit when peering VNet-Hub to VNet-Spoke
   - Set UseRemoteGateways when peering VNet-Spoke to VNet-Hub

Integration with Azure Standard Load Balancer

SCENARIO

Public LB in place for incoming traffic. Customer wishes to add Azure Firewall for data exfiltration protection.

When adding 0/0 to backend servers, things break

CHALLENGE

Asymmetric routing

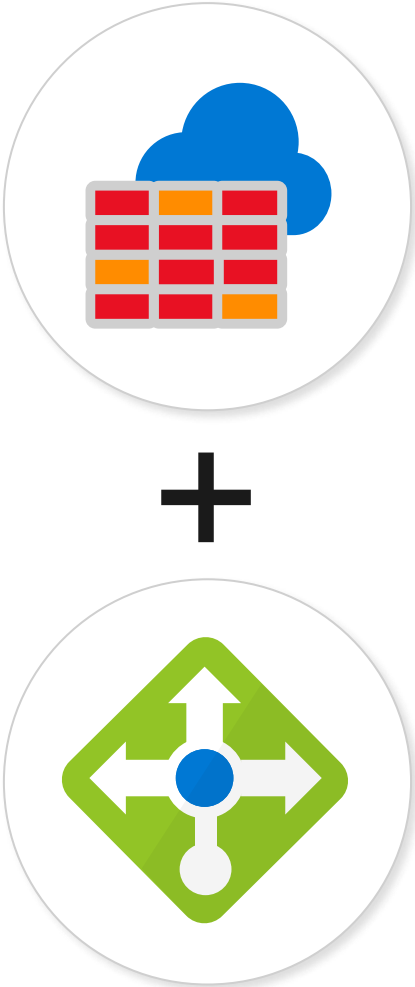
Packets take the inbound path via the Load Balancer but return via the Firewall. Since the Firewall is stateful, the response is dropped

SOLUTION

Make sure internet traffic to come in through your firewall's public IP address and use DNAT to forward traffic to the LB public IP

Use UDR to route packets destined for the firewall's public IP address via the internet

Consider using NSG to limit inbound internet access to backend servers to the firewall's public IP address



Integration with dedicated Azure Services in a VNet

SCENARIO

Dedicated Azure Service is deployed in a VNet. Customer wishes to add Azure Firewall for data exfiltration protection. When adding 0/0 to the service subnet, things break

CHALLENGE

Asymmetric Routing for incoming management traffic dropped by firewalls

Outbound dependencies for the dedicated services blocked by firewalls

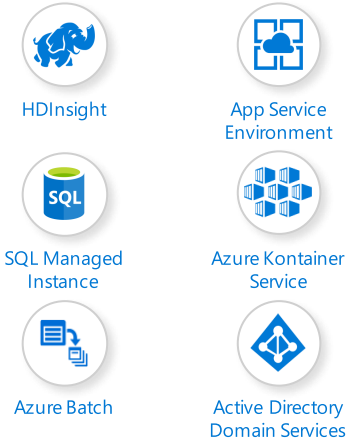
SOLUTION

Configure UDR to route management traffic directly to internet, bypassing the default route definition

Use FQDN Tags when possible

Inspect the firewall logs to gather the dependencies and use FQDN filtering to whitelist outbound traffic . Over time, this will be automated with more FQDN tags

+



Forcing Az Firewall traffic to on-prem or another NVA

SCENARIO

Customer configures default route on the firewall subnet to go to on-prem or another NVA for additional inspection

Things break

This can also happen as a result of BGP advertisement to send all traffic to on-prem

CHALLENGE

Azure Firewall is dependent on internet access to function properly. When sending Azure Firewall traffic on-premises or to another NVA, critical dependencies are not met

SOLUTION

Default route on the firewall subnet must have next hop set to Internet

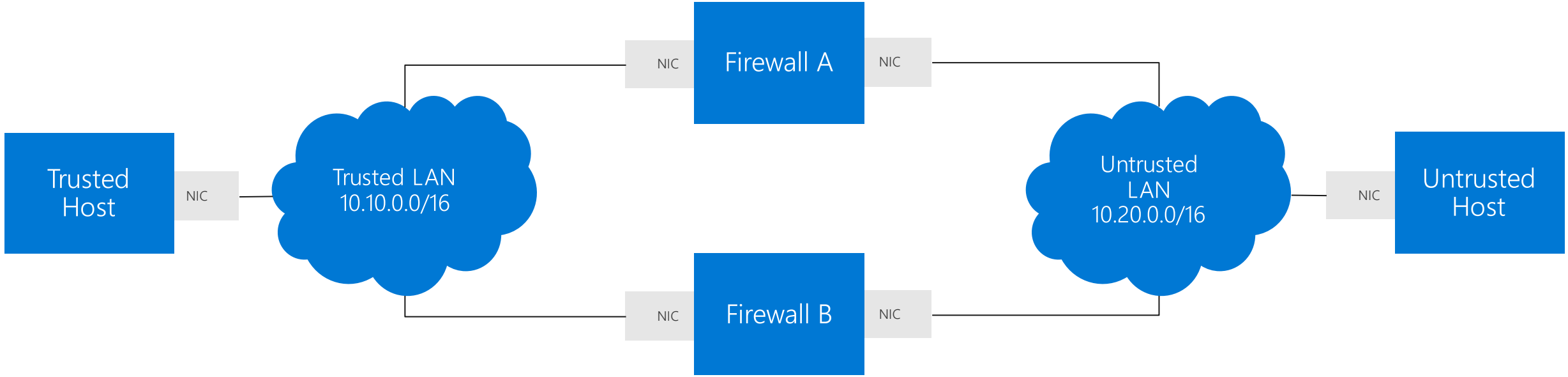
Other default routes are not supported and will be disabled in code

Chaining to another NVA is a roadmap item—no ETA yet

Setting specific routes on the firewall subnet is allowed

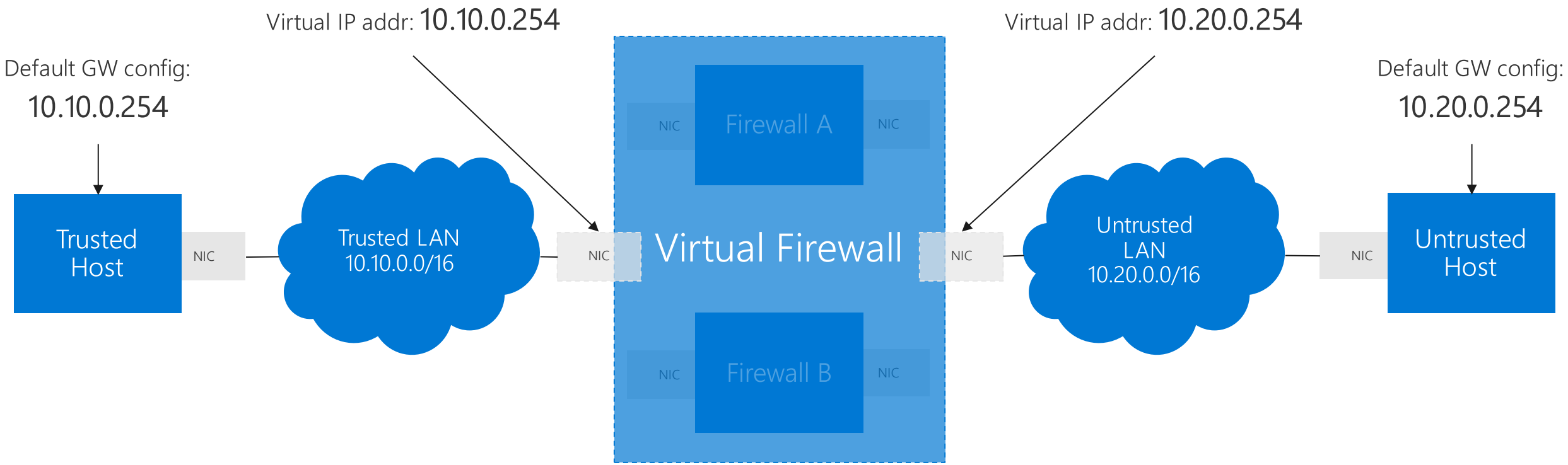
Rethinking HA protocols on Azure

Firewall clusters in physical networks
How can hosts select which firewall instance to use?



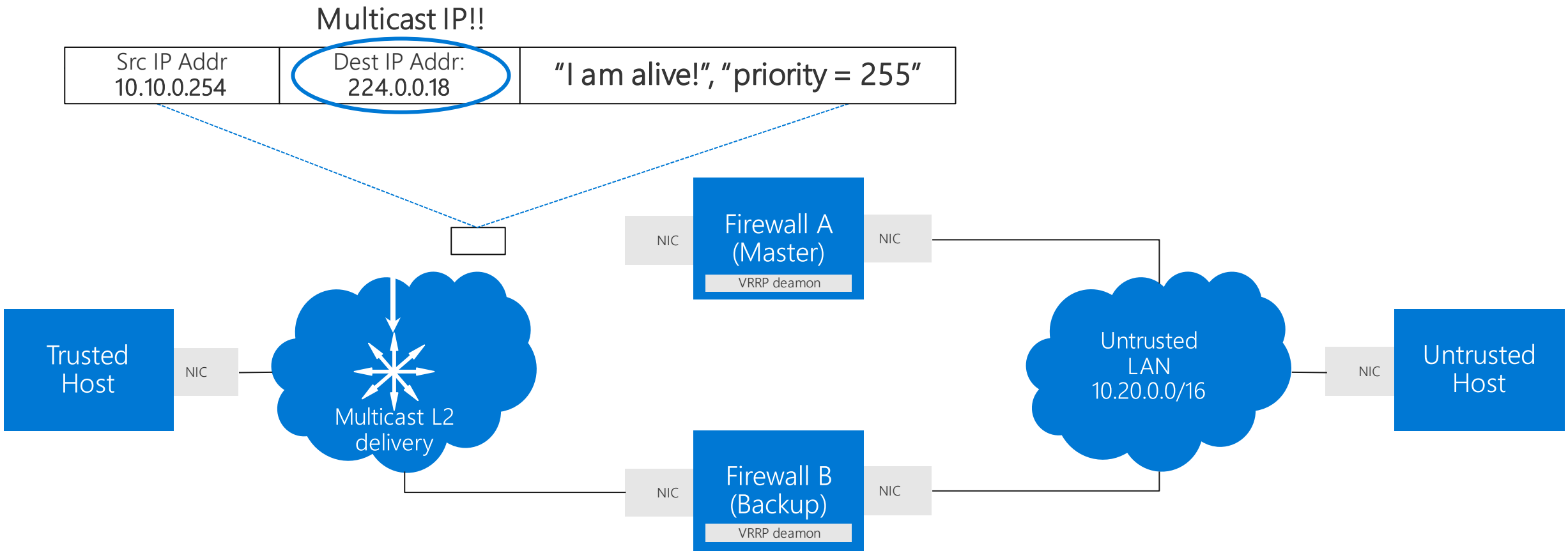
Running dynamic routing protocols on all hosts is impractical…

Router redundancy protocols (VRRP, HSRP)
Expose a single, highly available virtual device to hosts

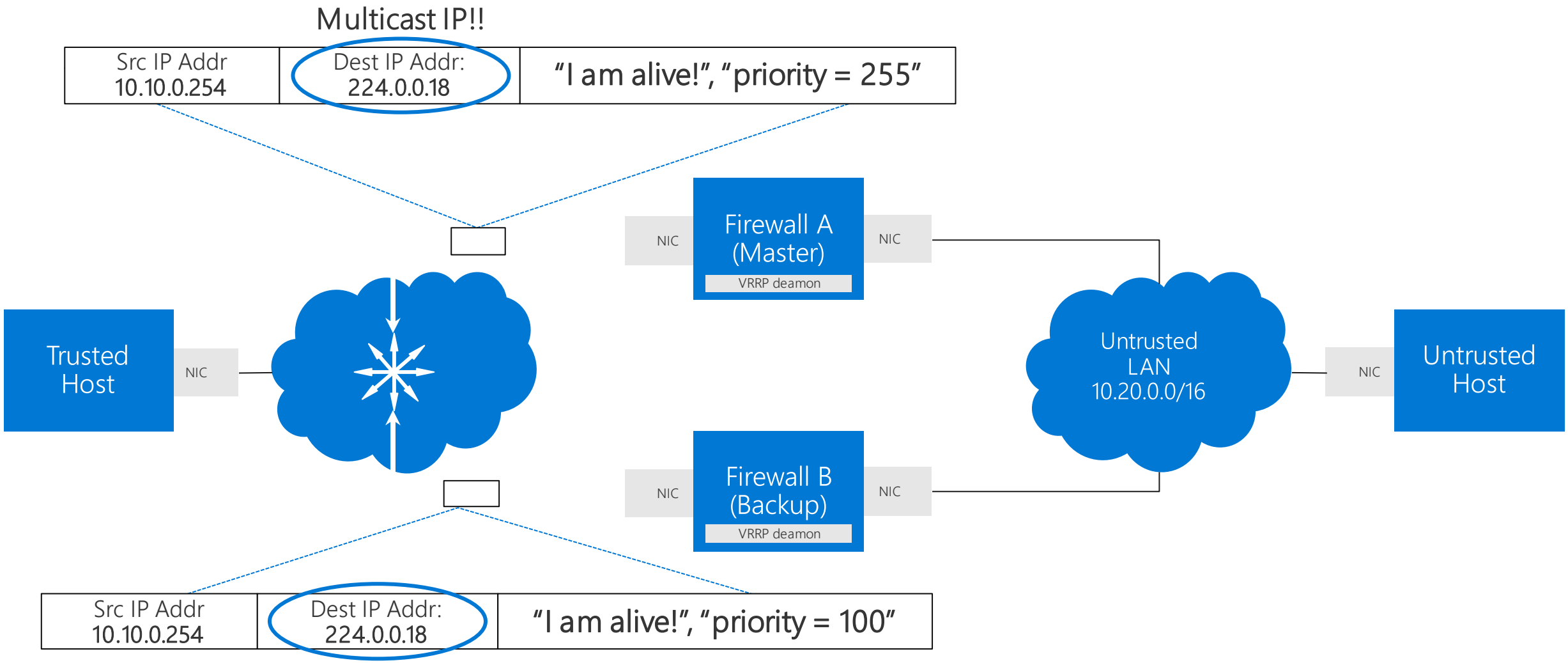


Default gateway statically set to virtual IP address. But how to implement the virtual device?

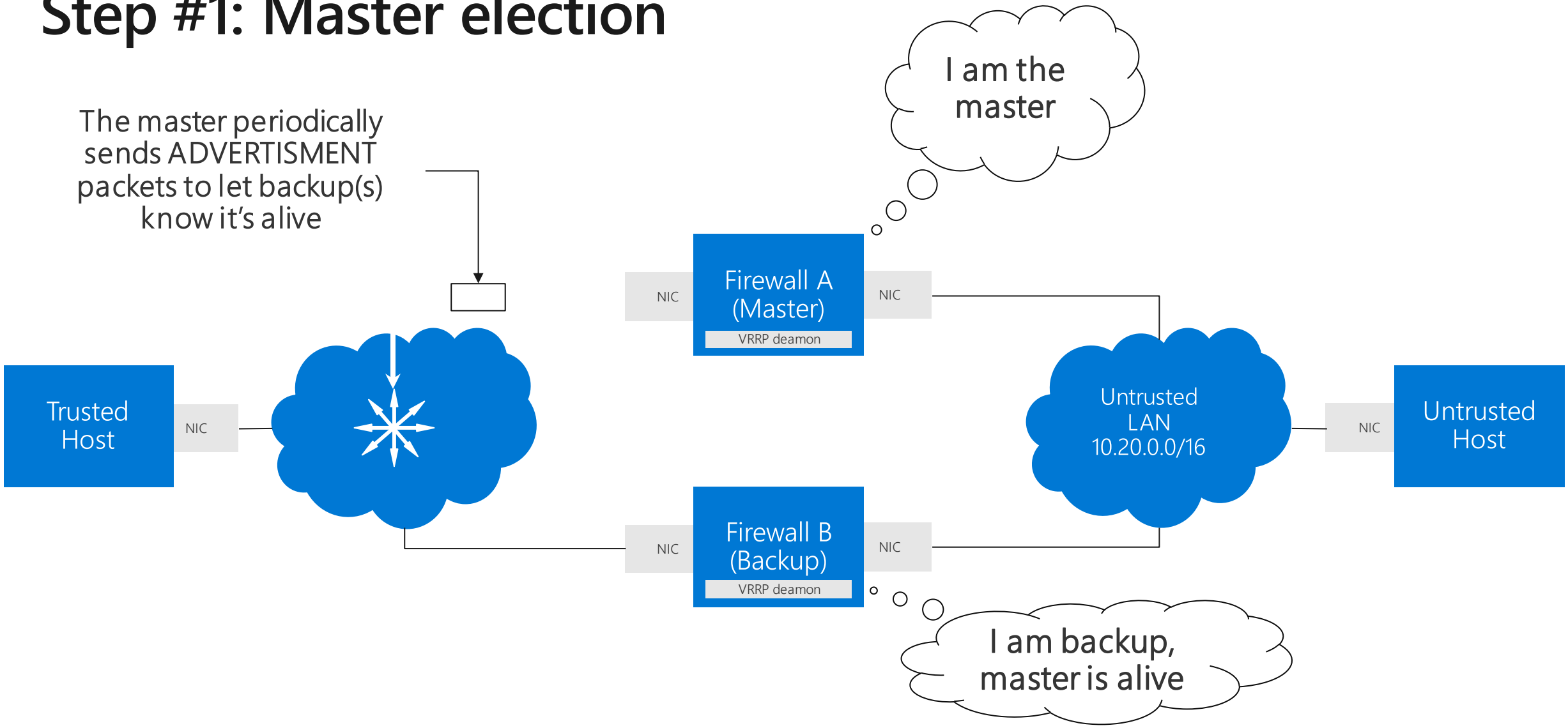
Router redundancy protocols (VRRP, HSRP)
Step #1: Master election



Router redundancy protocols (VRRP, HSRP)
Step #1: Master election

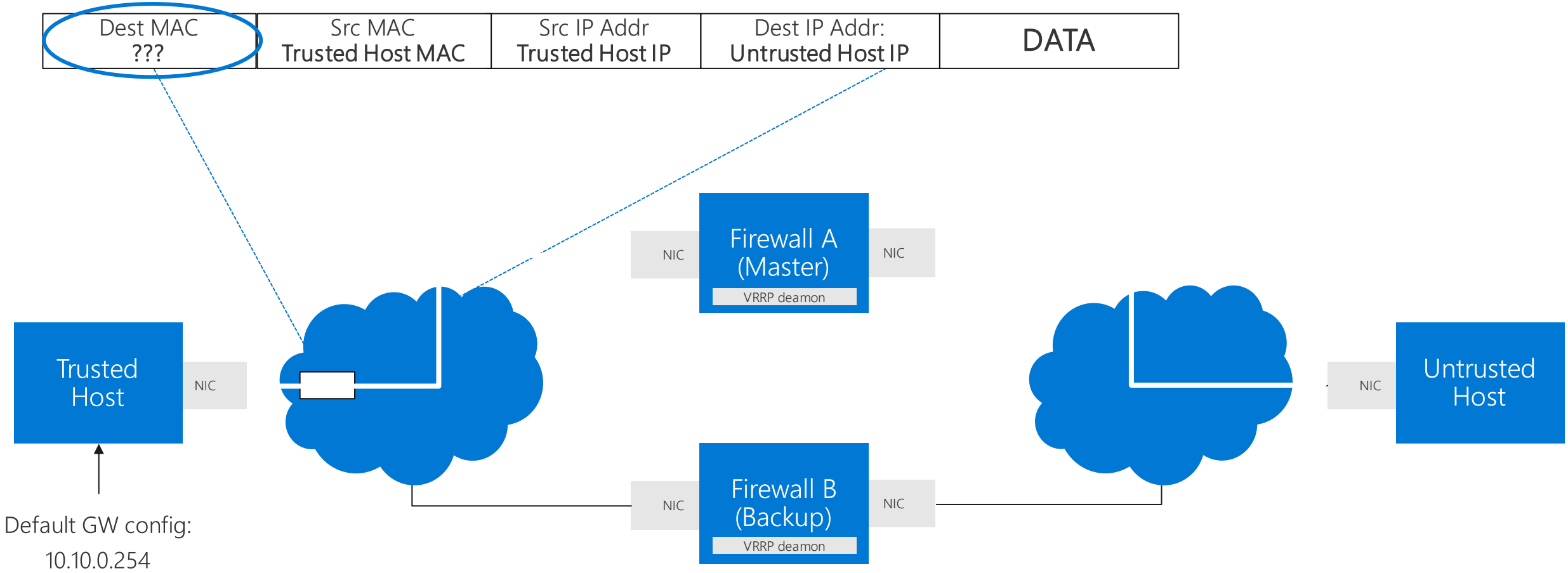


Router redundancy protocols (VRRP, HSRP)
Step #1: Master election



Router redundancy protocols (VRRP, HSRP)
Step #2: ARP resolution

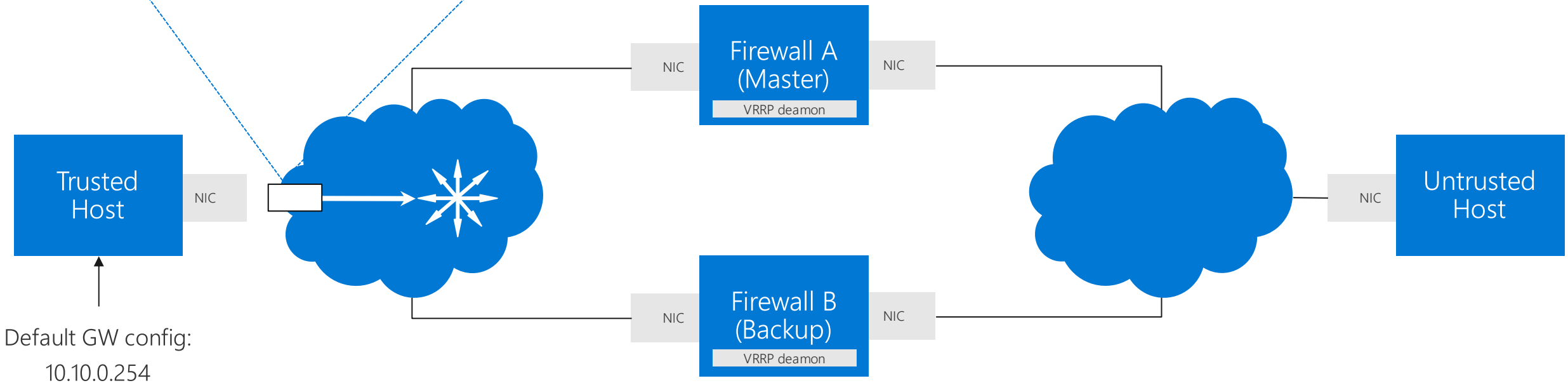
Trusted Host must find the firewall's MAC address!!



Router redundancy protocols (VRRP, HSRP)
Step #2: ARP resolution

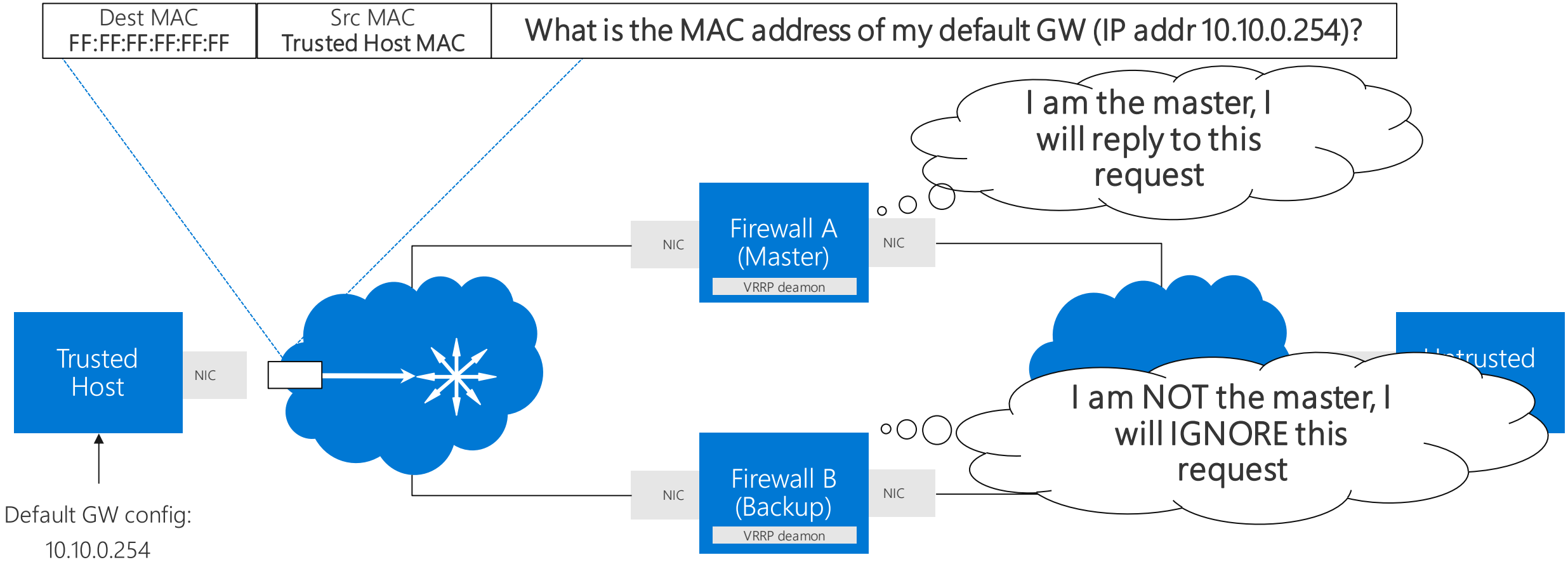
Trusted Host must find the firewall's MAC address!!

Dest MAC	Src MAC	
FF:FF:FF:FF:FF:FF	Trusted Host MAC	What is the MAC address of my default GW (IP addr 10.10.0.254)?

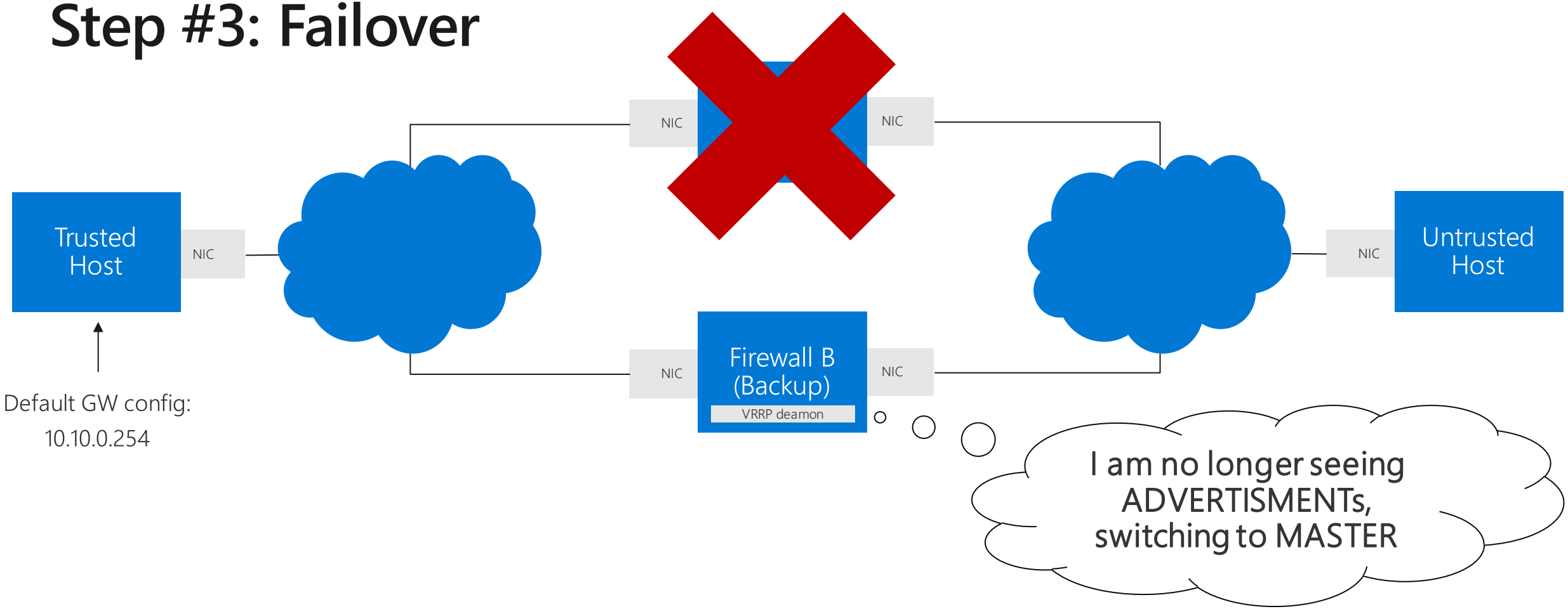


Router redundancy protocols (VRRP, HSRP)
Step #2: ARP resolution

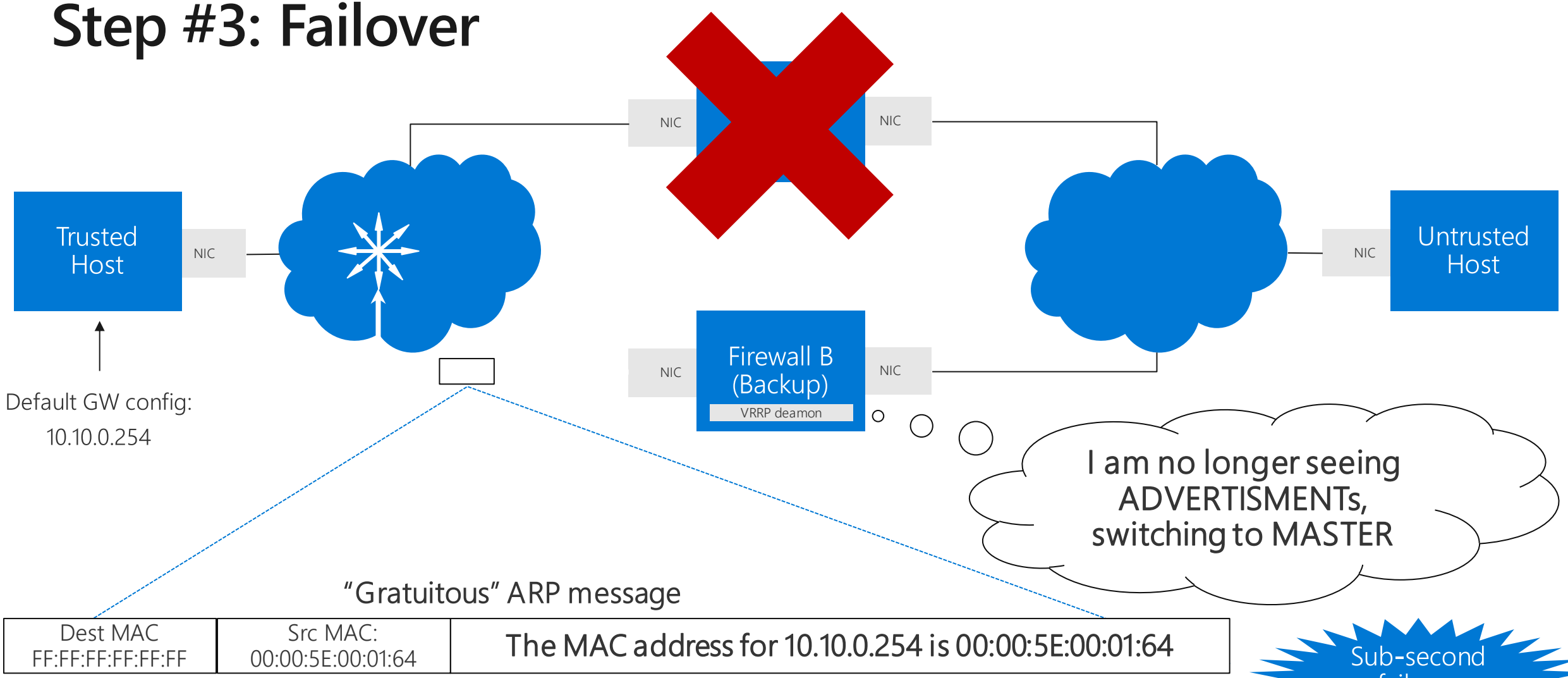
Trusted Host must find the firewall's MAC address!!



Router redundancy protocols (VRRP, HSRP)
Step #3: Failover



Router redundancy protocols (VRRP, HSRP)
Step #3: Failover



VRRP and HSRP do NOT work in Azure

Session Takeaway

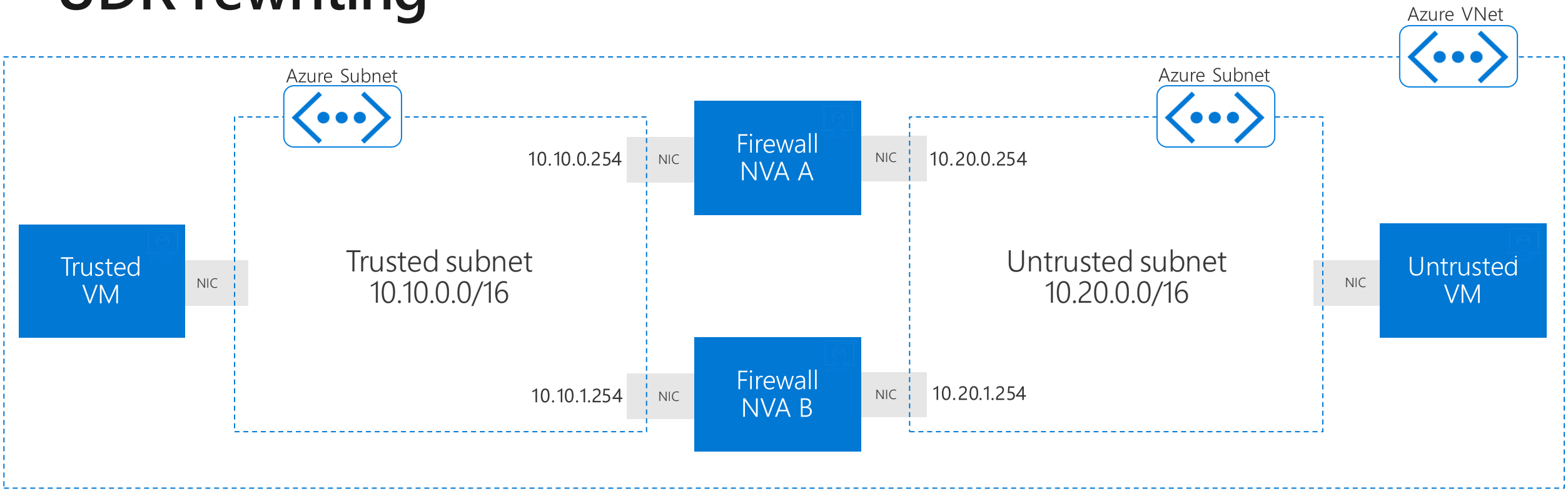
IP multicast addresses cannot be used in Azure Virtual Networks

Broadcast ARP messages generated by VMs are intercepted by Azure's network virtualization stack and NOT forwarded to other VMs

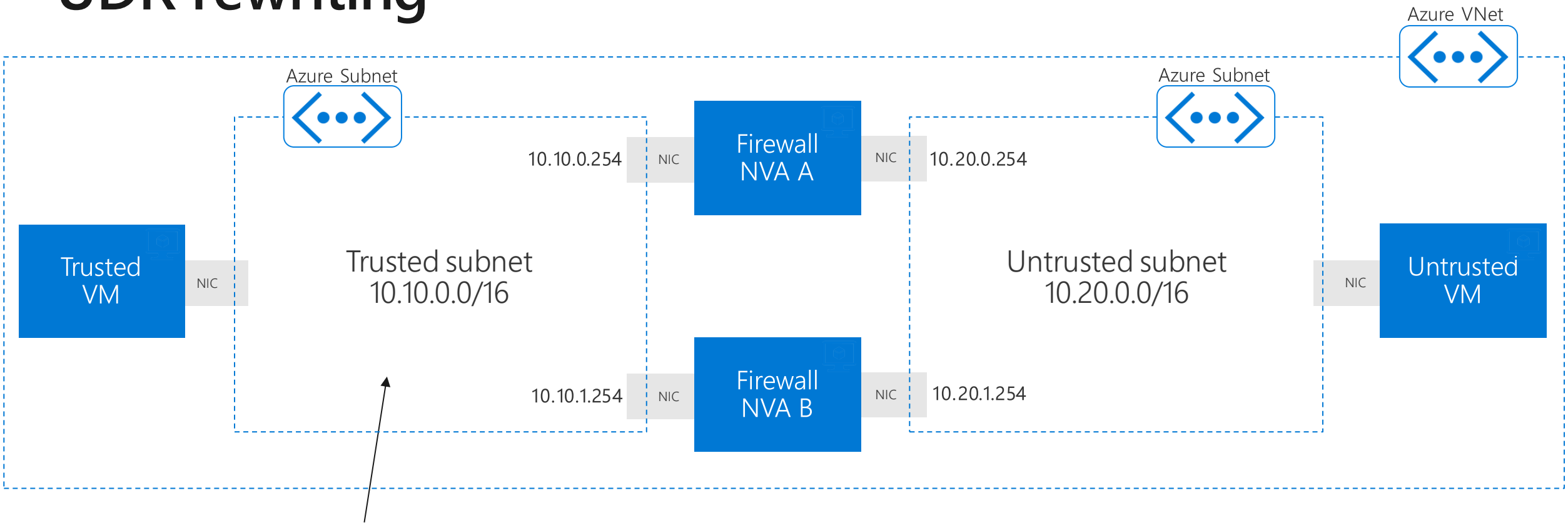
In Azure, firewall clusters cannot be implemented using traditional router redundancy protocols such as VRRP and HSRP

Active/Standby with UDR rewriting

UDR rewriting

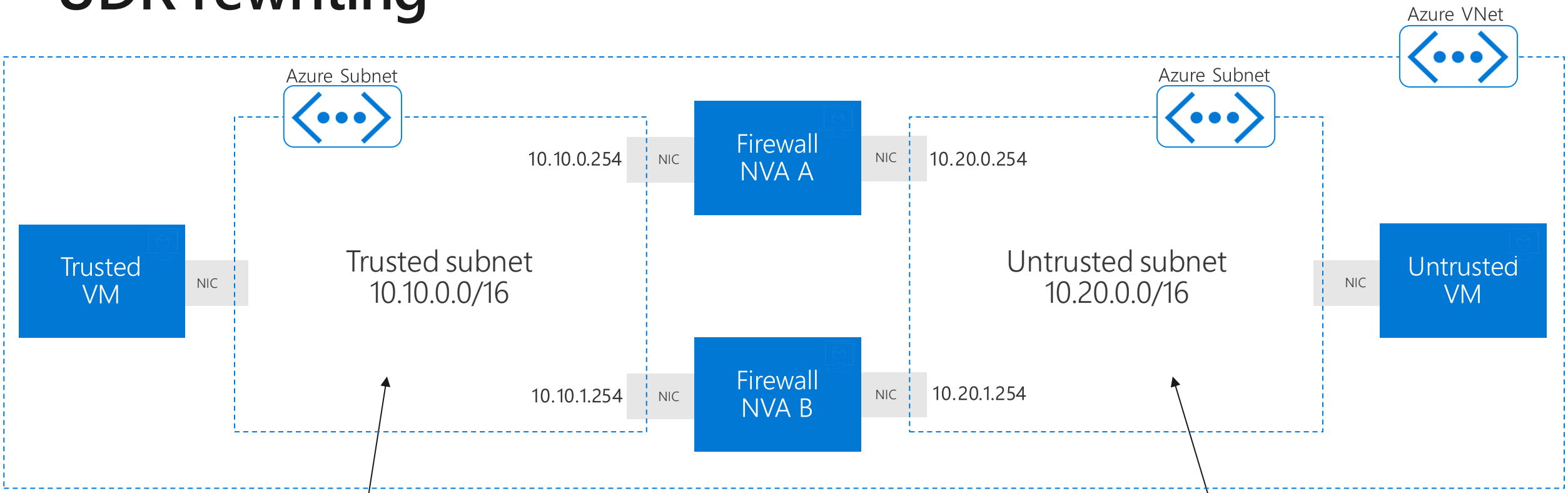


UDR rewriting



Custom Route Table for Trusted Subnet		
Destination	**Next Hop Type**	**Next Hop IP**
10.10.0.0/16	**Virtual Network**	**---**
10.20.0.0/16	**Virtual Appliance**	**10.10.0.254**

UDR rewriting



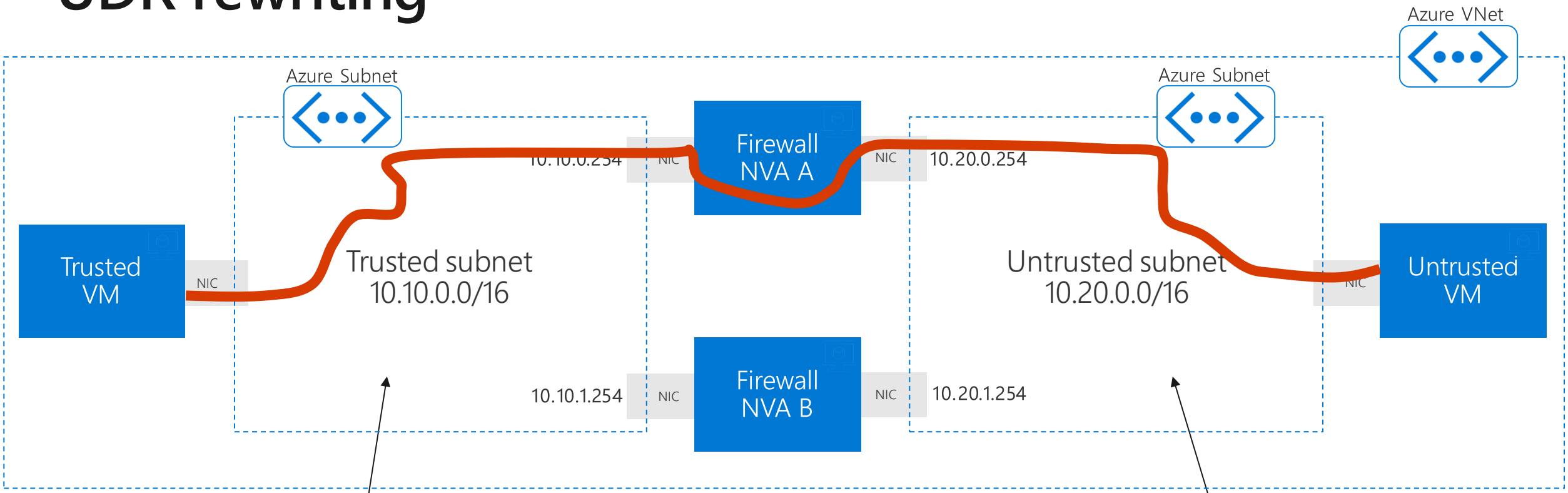
Custom Route Table for Trusted Subnet

Destination	Next Hop Type	Next Hop IP
10.10.0.0/16	**Virtual Network**	**---**
10.20.0.0/16	**Virtual Appliance**	**10.10.0.254**

Custom Route Table for Untrusted Subnet

Destination	Next Hop Type	Next Hop IP
10.10.0.0/16	**Virtual Appliance**	**10.20.0.254**
10.20.0.0/16	**Virtual Network**	**---**

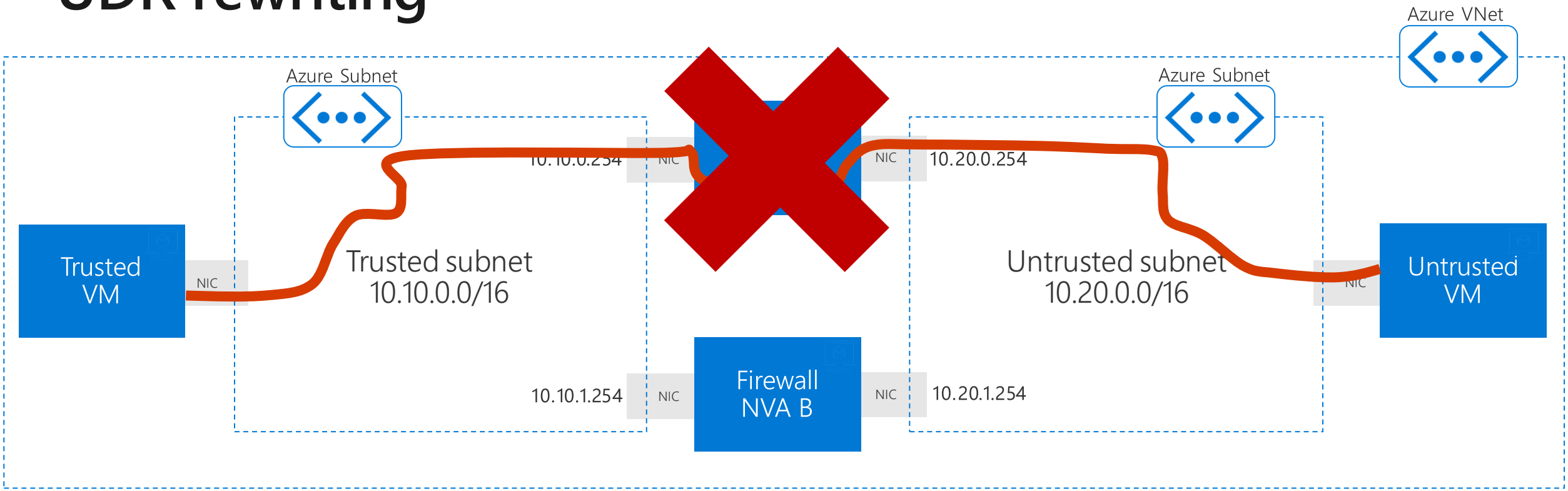
UDR rewriting



Custom Route Table for Trusted Subnet		
Destination	**Next Hop Type**	**Next Hop IP**
10.10.0.0/16	**Virtual Network**	**---**
10.20.0.0/16	**Virtual Appliance**	**10.10.0.254**

Custom Route Table for Untrusted Subnet		
Destination	Next Hop Type	Next Hop IP
10.10.0.0/16	**Virtual Appliance**	**10.20.0.254**
10.20.0.0/16	**Virtual Network**	**---**

UDR rewriting

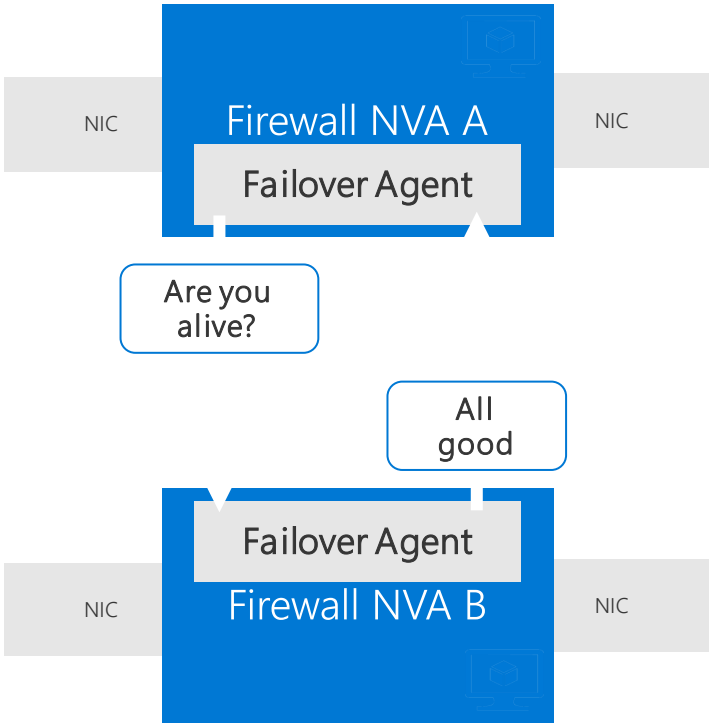


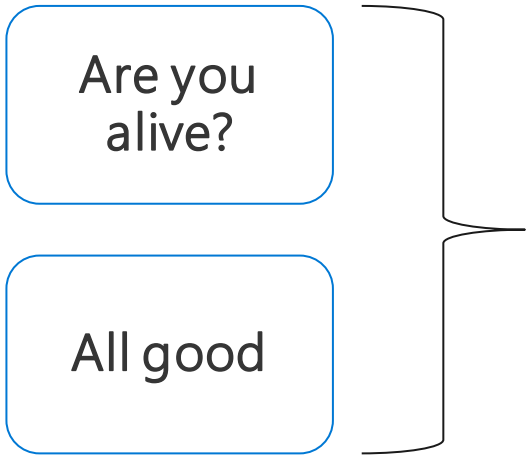
When active NVA goes offline, failover logic must:

Detect the failure

Access Azure's management API and update custom route tables

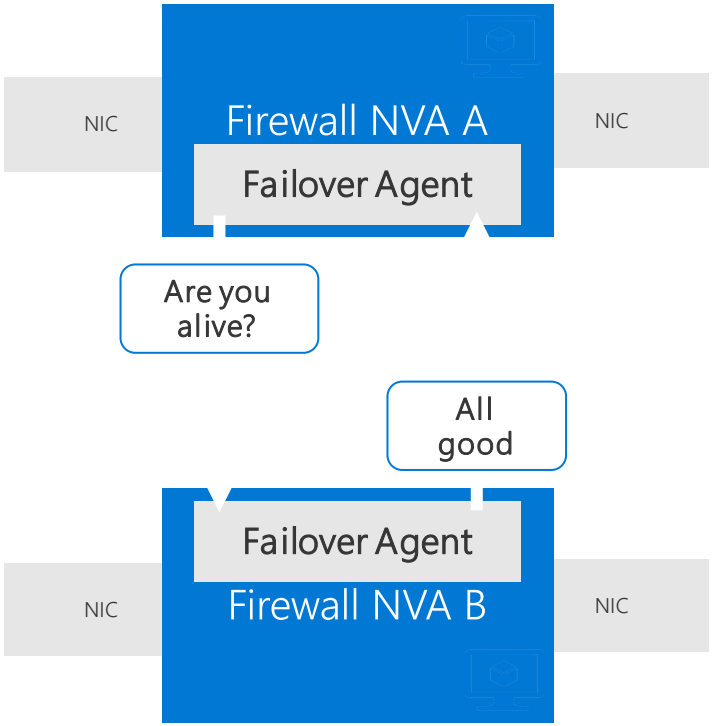
UDR rewriting
Failover logic running on the NVA's

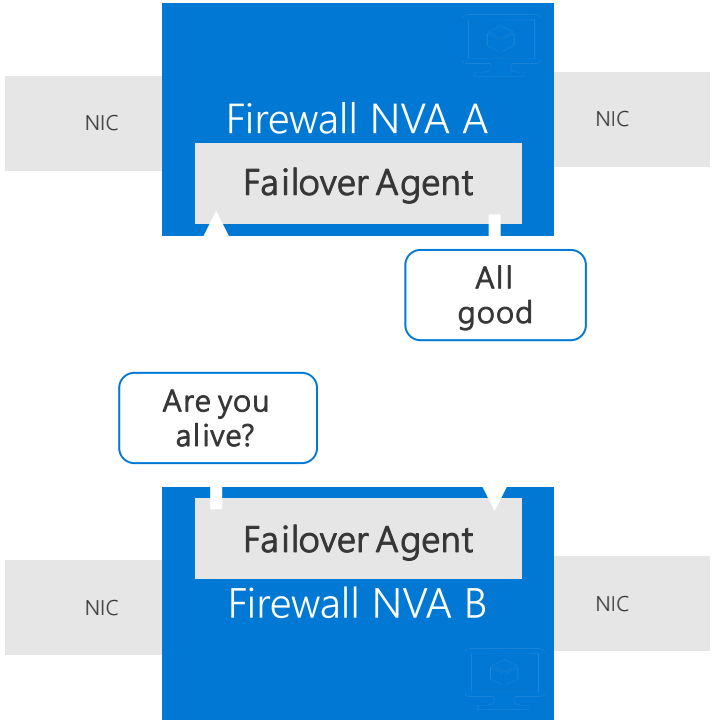


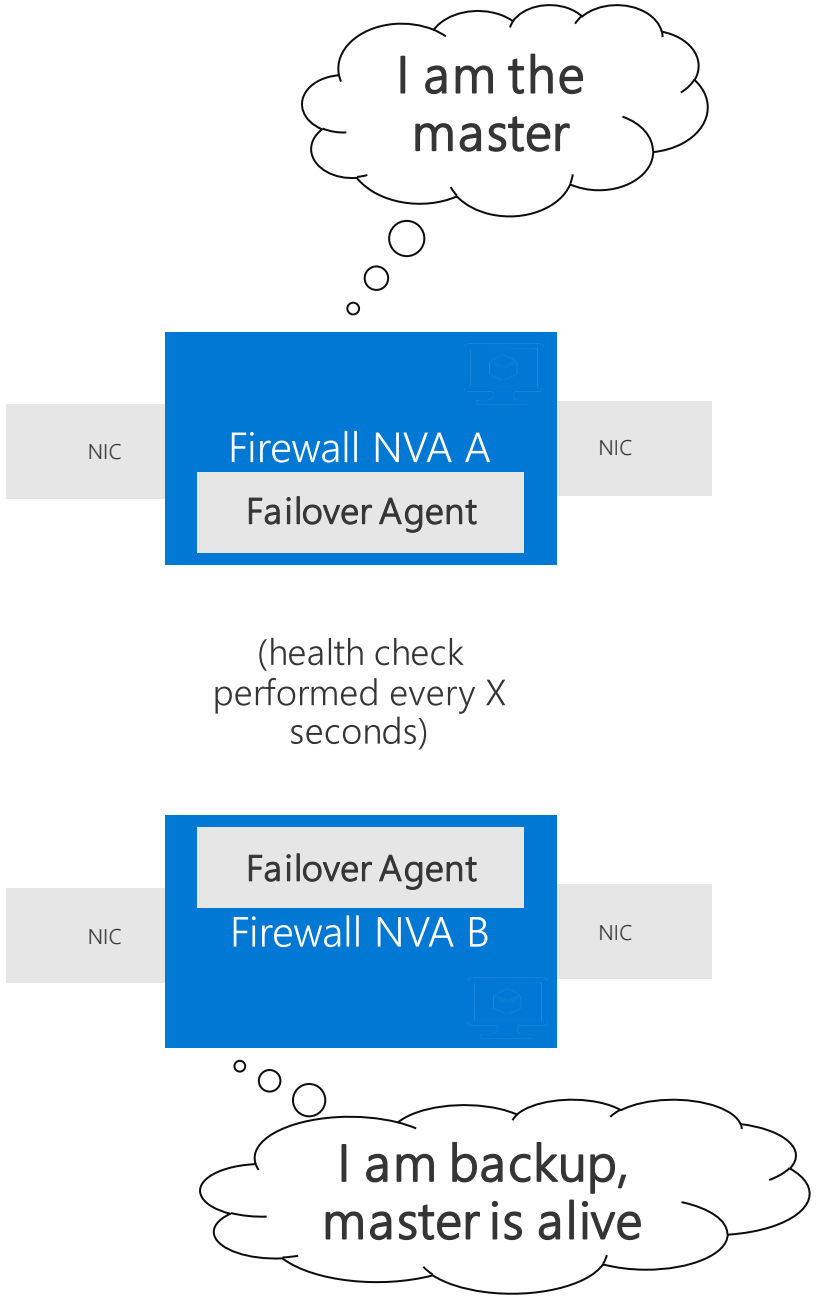


Usually TCP 3-way handshake to peer's management port (e.g. SSH TCP/22)

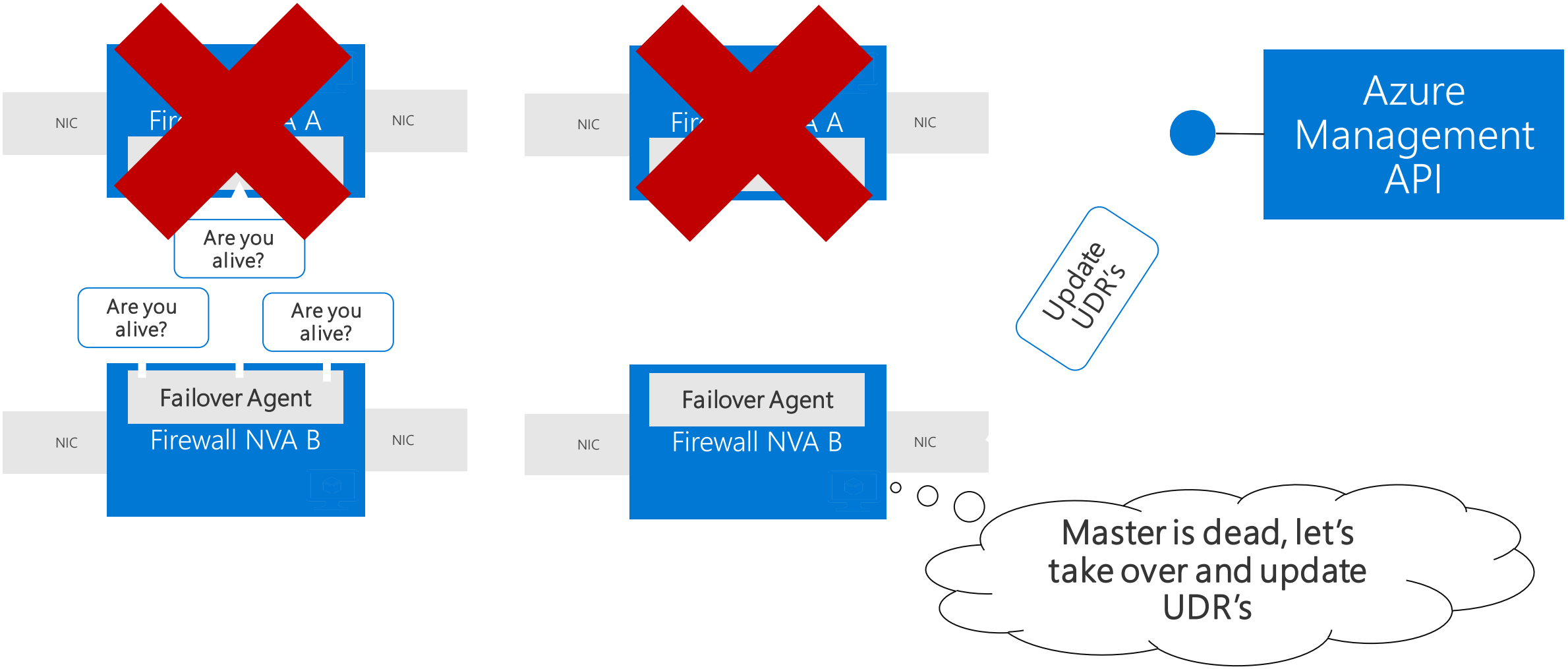
UDR rewriting
Failover logic running on the NVA's



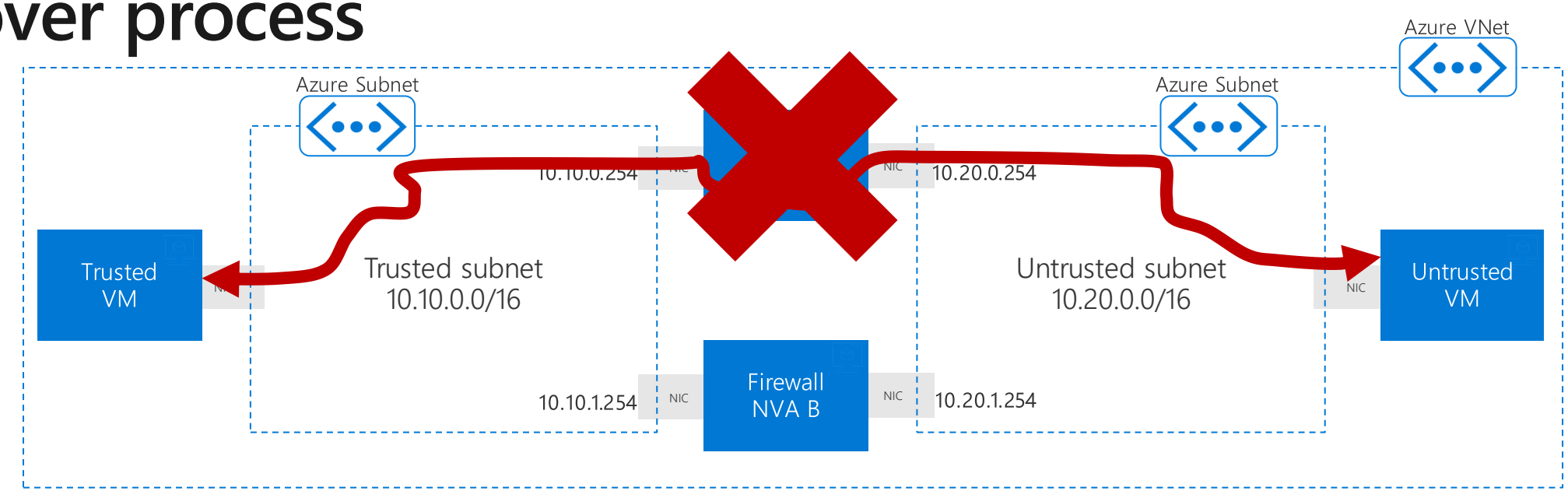




UDR rewriting
Failover process



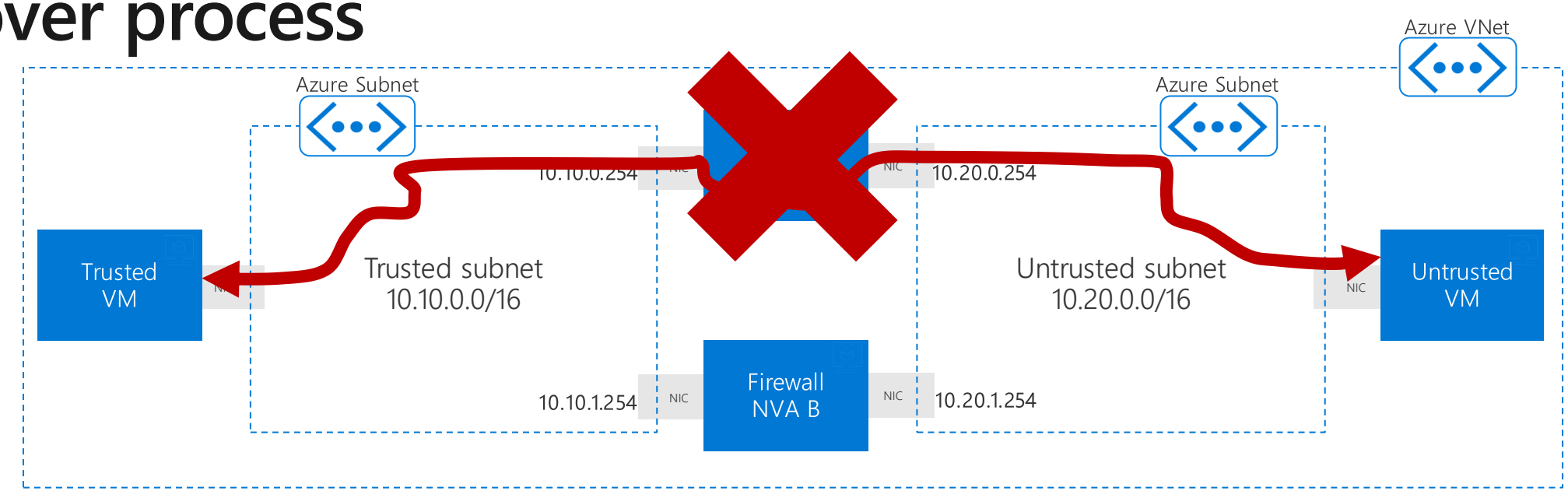
UDR rewriting
Failover process



Custom Route Table for Trusted Subnet		
Destination	**Next Hop Type**	**Next Hop IP**
10.10.0.0/16	**Virtual Network**	**---**
10.20.0.0/16	**Virtual Appliance**	**10.10.0.254**

Custom Route Table for Untrusted Subnet		
Destination	Next Hop Type	Next Hop IP
10.10.0.0/16	**Virtual Appliance**	**10.20.0.254**
10.20.0.0/16	**Virtual Network**	**---**

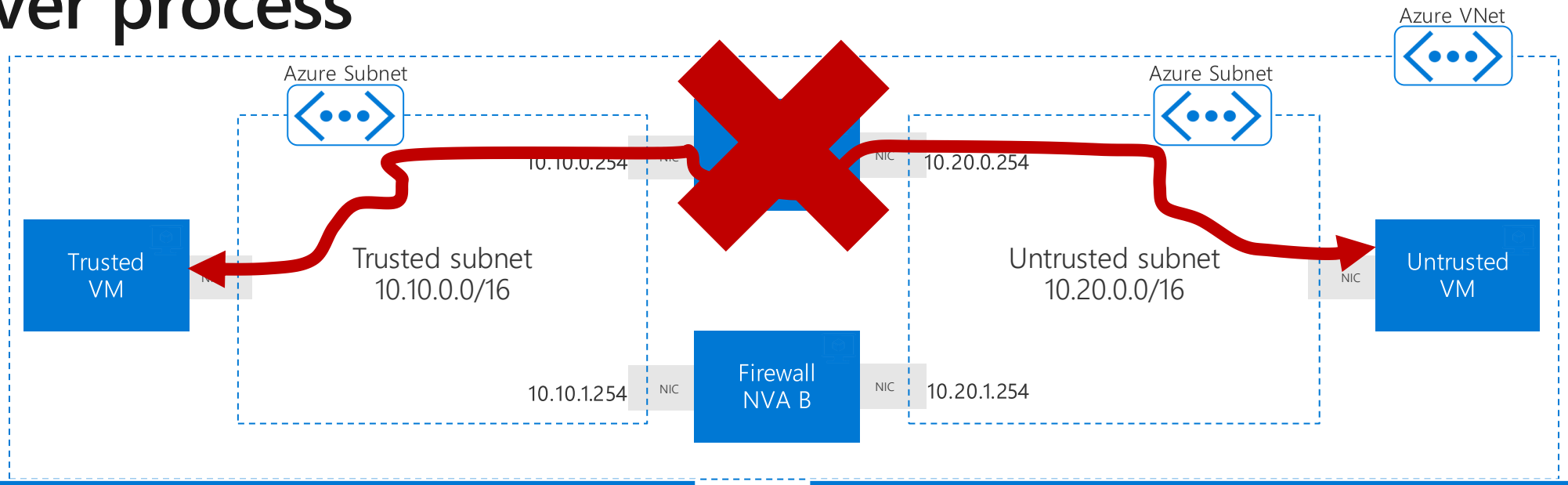
UDR rewriting
Failover process



Custom Route Table for Trusted Subnet		
Destination	**Next Hop Type**	**Next Hop IP**
10.10.0.0/16	**Virtual Network**	**---**
~~**10.20.0.0/16**~~	~~**Virtual Appliance**~~	~~**10.10.0.254**~~

Custom Route Table for Untrusted Subnet		
Destination	Next Hop Type	Next Hop IP
~~**10.10.0.0/16**~~	~~**Virtual Appliance**~~	~~**10.20.0.254**~~
10.20.0.0/16	**Virtual Network**	**---**

UDR rewriting
Failover process



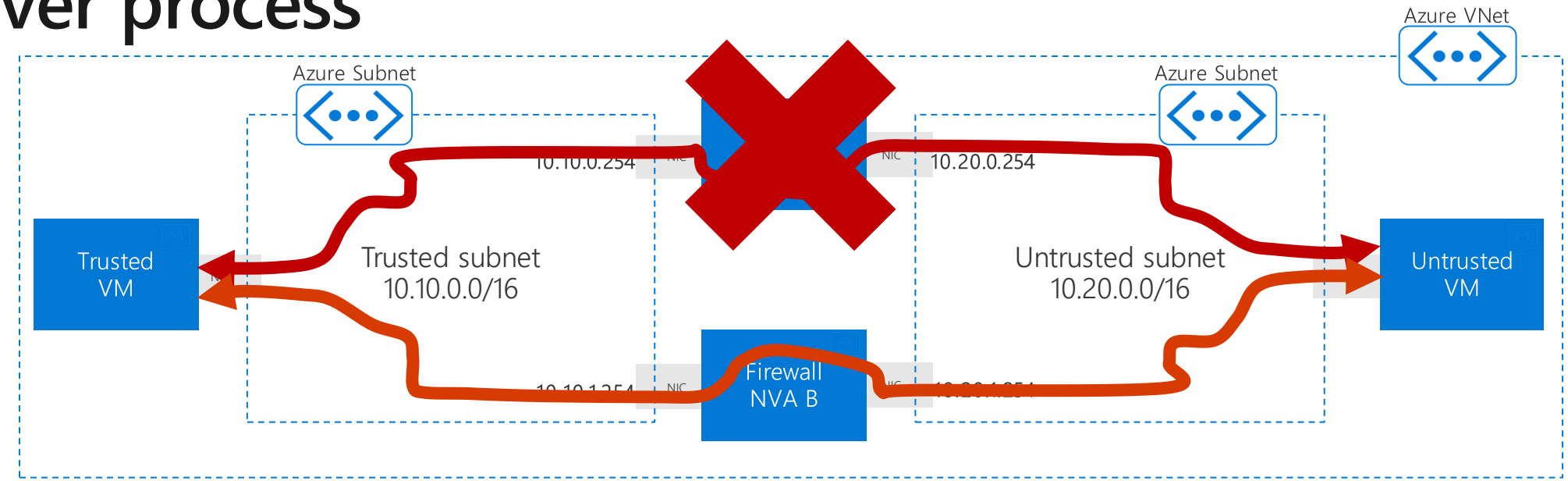
Custom Route Table for Trusted Subnet

Destination	Next Hop Type	Next Hop IP
10.10.0.0/16	Virtual Network	---
~~10.20.0.0/16~~	~~Virtual Appliance~~	~~10.10.0.254~~
10.20.0.0./16	Virtual Appliance	10.10.1.254

Custom Route Table for Untrusted Subnet

Destination	Next Hop Type	Next Hop IP
~~10.10.0.0/16~~	~~Virtual Appliance~~	~~10.20.0.254~~
10.10.0.0./16	Virtual Appliance	10.20.1.254
10.20.0.0/16	Virtual Network	---

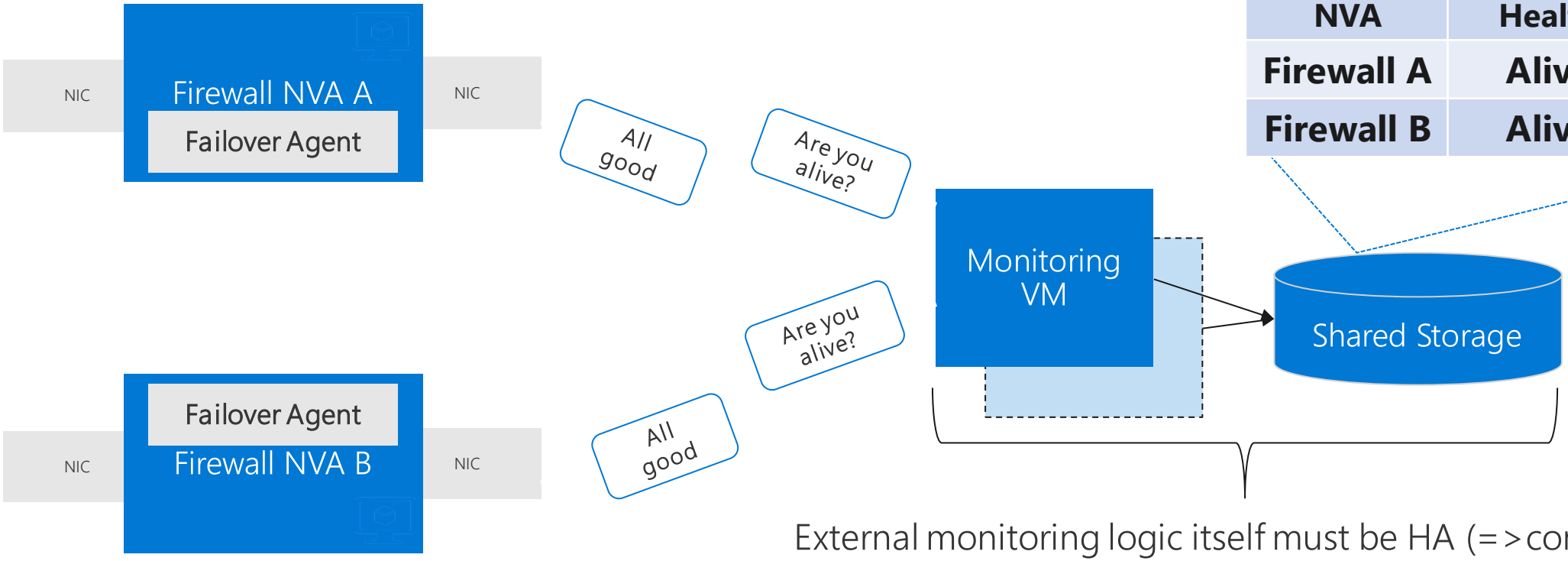
UDR rewriting
Failover process



Custom Route Table for Trusted Subnet		
Destination	**Next Hop Type**	**Next Hop IP**
10.10.0.0/16	**Virtual Network**	**---**
~~10.20.0.0/16~~	**~~Virtual Appliance~~**	**~~10.10.0.254~~**
10.20.0.0./16	**Virtual Appliance**	**10.10.1.254**

Custom Route Table for Untrusted Subnet		
Destination	Next Hop Type	Next Hop IP
~~10.10.0.0/16~~	**~~Virtual Appliance~~**	**~~10.20.0.254~~**
10.10.0.0./16	**Virtual Appliance**	**10.20.1.254**
10.20.0.0/16	**Virtual Network**	**---**

UDR rewriting
Failover logic running on external VM(s)

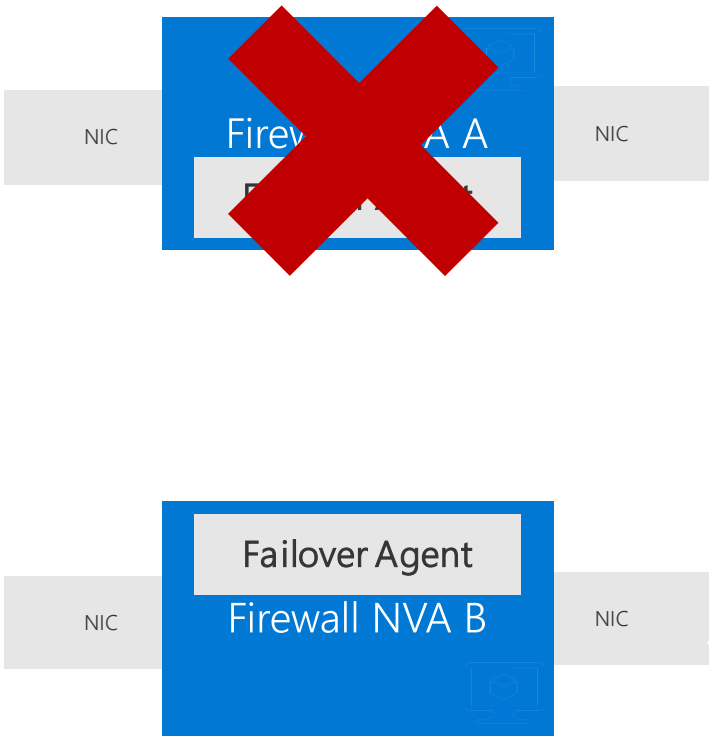


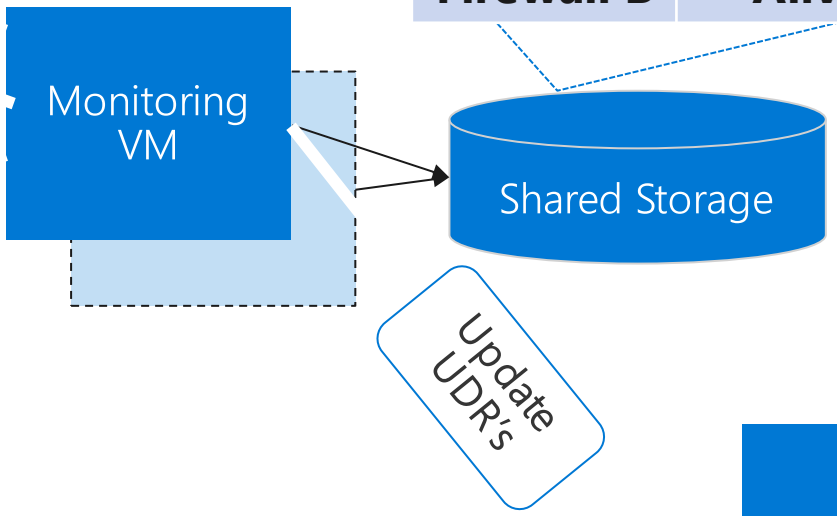
Cluster State		
NVA	Health	Active
Firewall A	Alive	Yes
Firewall B	Alive	No

External monitoring logic itself must be HA (=>complexity)

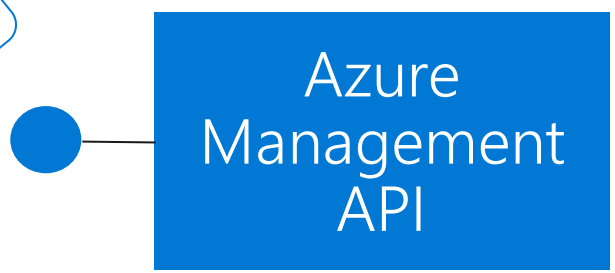
Azure Doc provides guidance on how to implement HA external monitoring logic with Zookeeper (link)

UDR rewriting
Failover logic running on external VM(s)





Cluster State		
NVA	Health	Active
Firewall A	~~Alive~~ Dead	~~Yes~~ No
Firewall B	Alive	~~No~~ Yes



UDR rewriting Failover time

Failover time	=	Time to detect failure in master NVA (depends on probe freq)	+	Time to apply new UDR's
50 – 100+ seconds	=	< 10 seconds (probe every 5s)	+	40 – 100+ seconds

Way too much for many enterprise workloads!! (VRRP/HSRP achieve sub-second failover time)

UDR rewriting
Vendor support

Vendor	Product (marketplace link)	Failover logic	Documentation Available?	Notes
Barracuda	NextGen Firewall F-Series (link)	Running within NVA	Yes (link)	Single NIC, Single IP
Checkpoint	vSEC for Azure (link)	Running within NVA	Yes (link)	Dual/Multiple NIC
Cisco	ASAv (link)	Running within NVA	Yes(pdf, video)	Dual/Multiple NIC
F5	Big-IP LTM (link)	Running within NVA	Yes (link)	
Any	Any	Running on Zookeeper	Yes (link)	Non-product-specific guidance provided by MS

Internal segmentation with UDR rewriting

Session Takeaway

Requires complex monitoring logic, running either on the NVA's or on external VM's

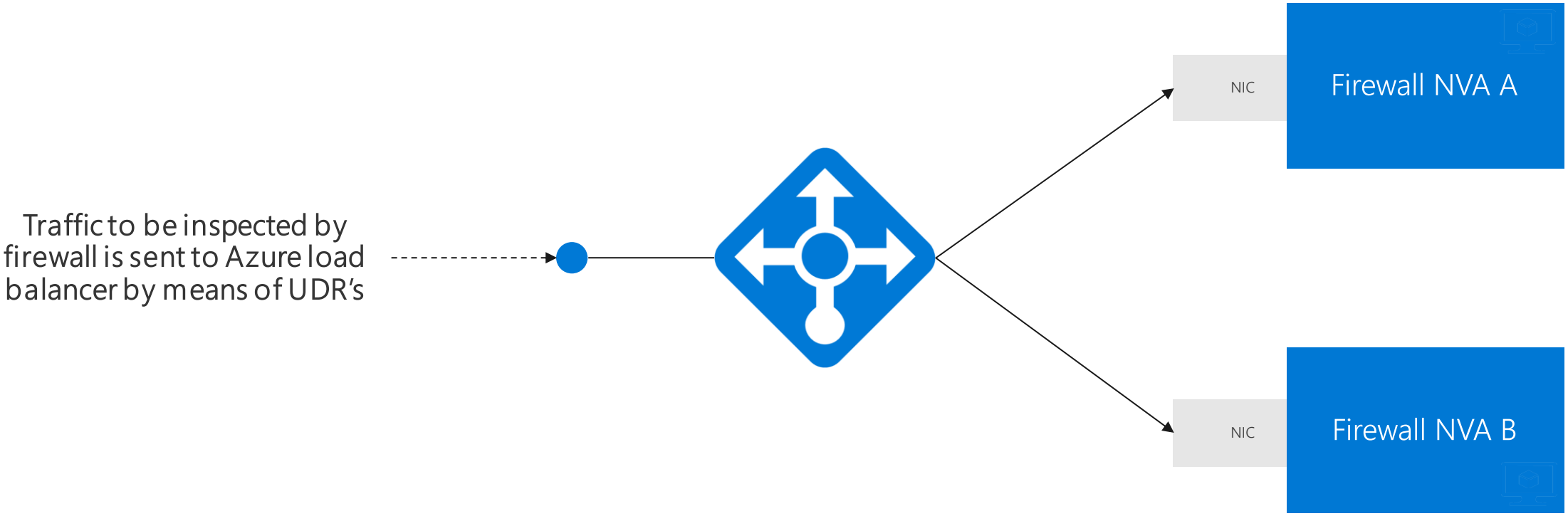
Failover time ranges from 50 to hundreds of seconds – too long for many enterprise scenarios

Active/Standby only, limited scalability

No longer the recommended HA option

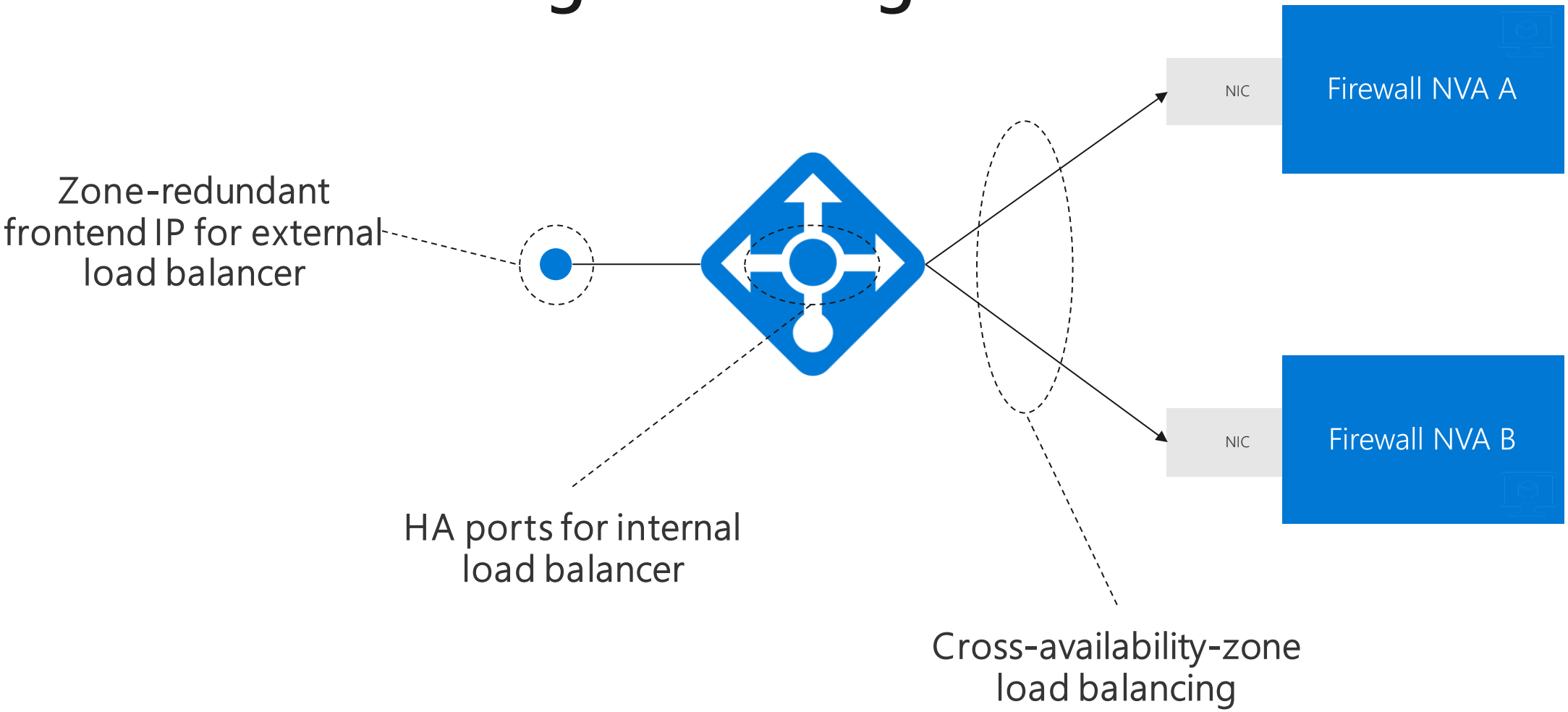
Active/Active with Azure load balancer

Using Azure load balancer with NVA's



- In each protected subnet, UDR's are defined with next hop = load balancer IP
- No need to update UDR's if a NVA fails

Using Azure load balancer with NVA's Standard SKU is a game changer!

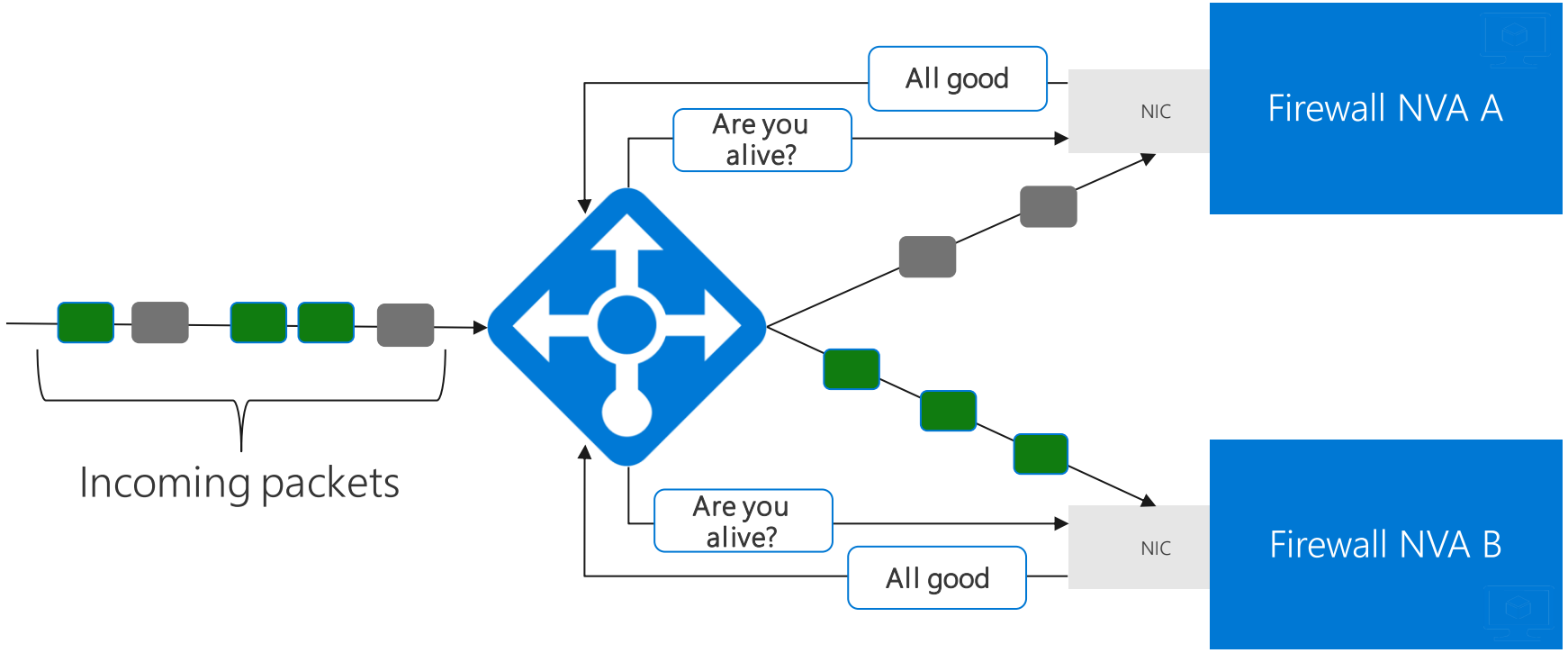


Using Azure load balancer with NVA's

Session Takeaway

Load balancer standard SKU is needed to implement the active/active configurations discussed next

Active/Standby vs. Active/Active
Why Active/Active is the best option?



Active/Active: Azure load balancer monitors the instances' health

No need for HA agents/external monitoring logic

Active/Active with Azure load balancer Failover time

Failover time	=	Time needed by load balancer to detect NVA failure	+	Time load balancer reconfiguration
12 - 25s	=	< 10s (probe every 5s)		10s – 15s

Still higher than VRRP/HSRP, but much better than Active/Standby with UDR rewriting!

Why active/active?
Active/Standby vs. Active/Active (or N-active)

Session Takeaway

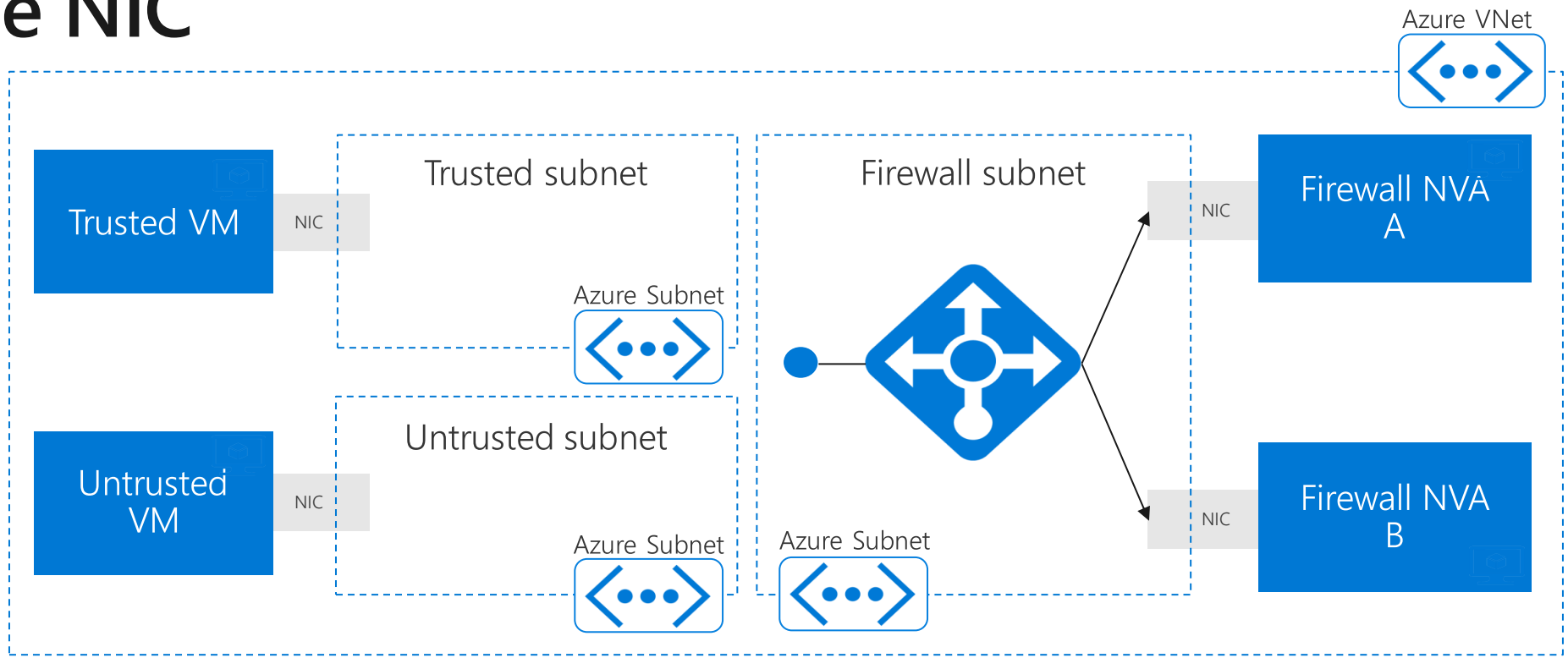
The platform supports both active/standby and active/active designs

Active/active is *less complex* (no HA agents needed on the NVAs)

Active/active provides *better scalability*

When using load balancers, active/active (or N-active) configurations should be preferred

How many NICs/load balancers?
Single NIC

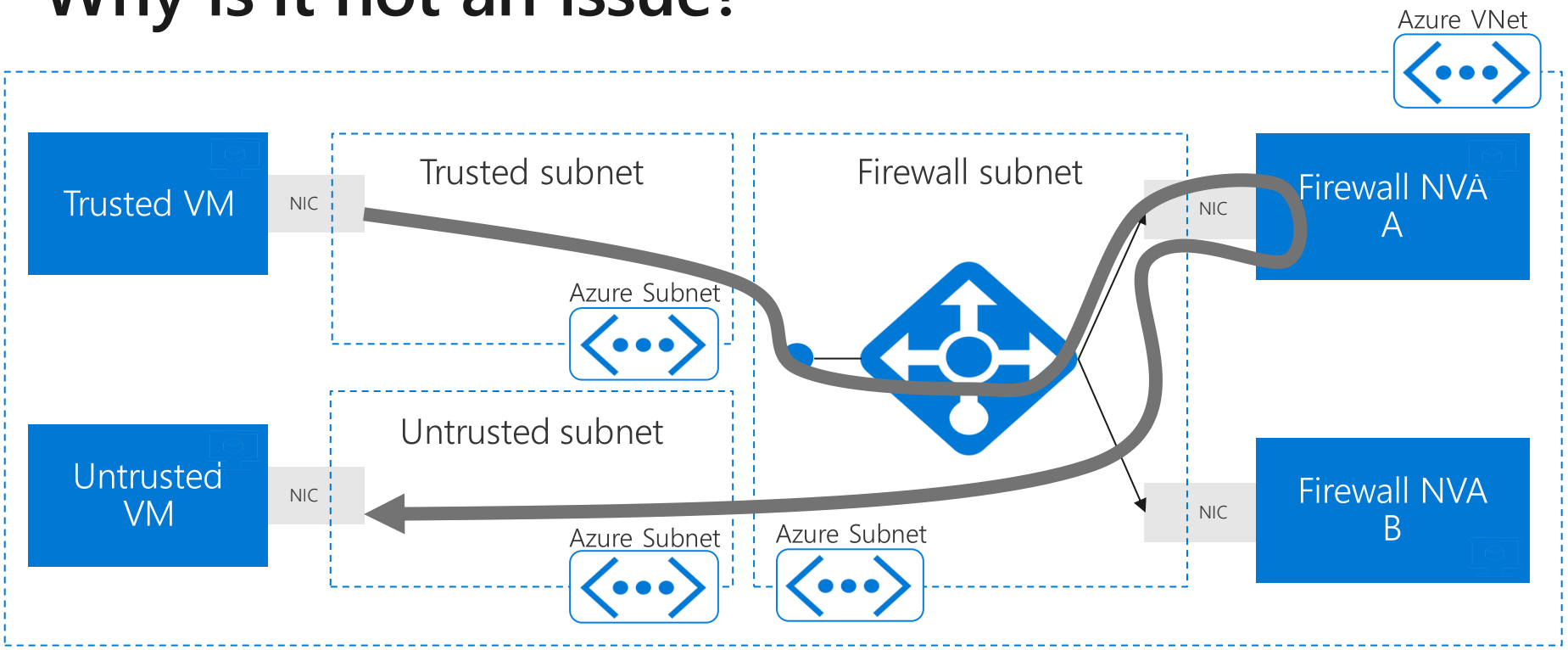


NVAs have no NICs attached to protected subnets

Single load balancer frontend for traffic from both “Trusted” and “Untrusted” subnets

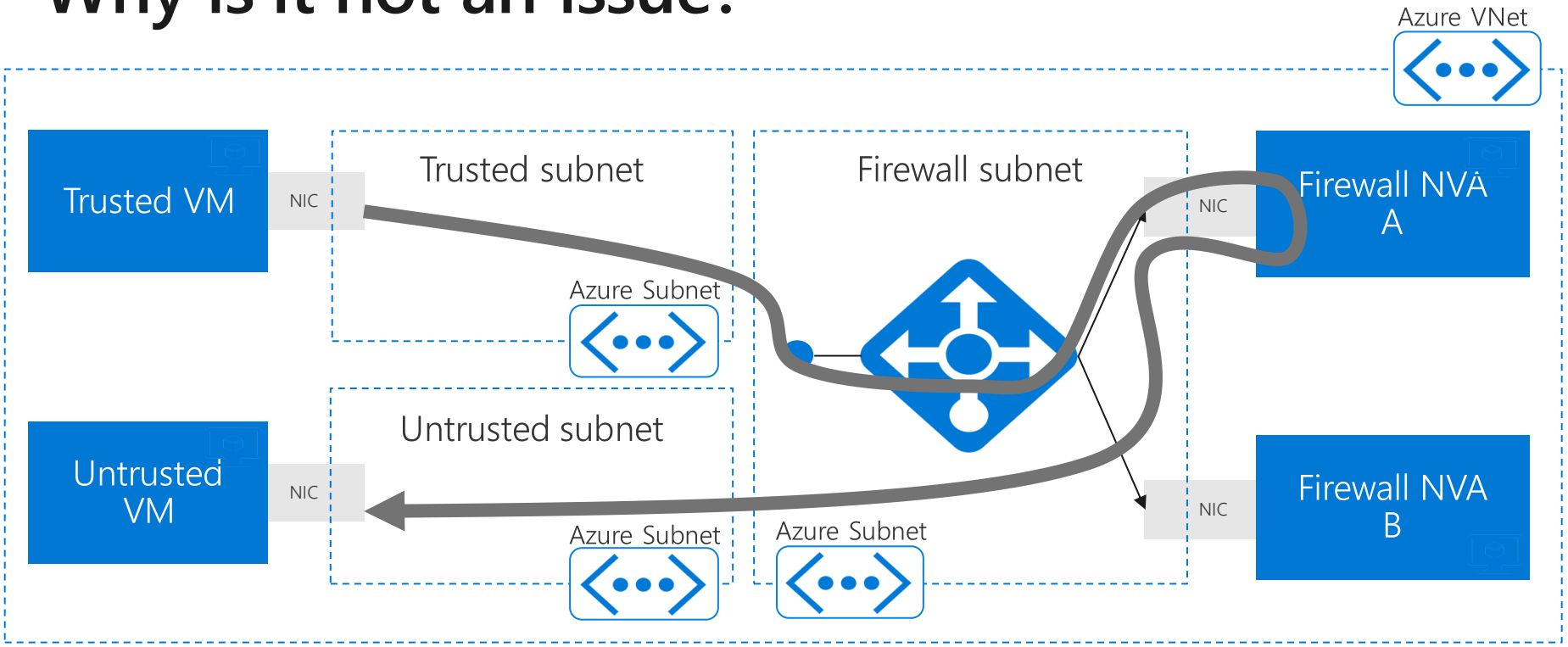
The same “anyport” load balancing rule distributes traffic from both subnets

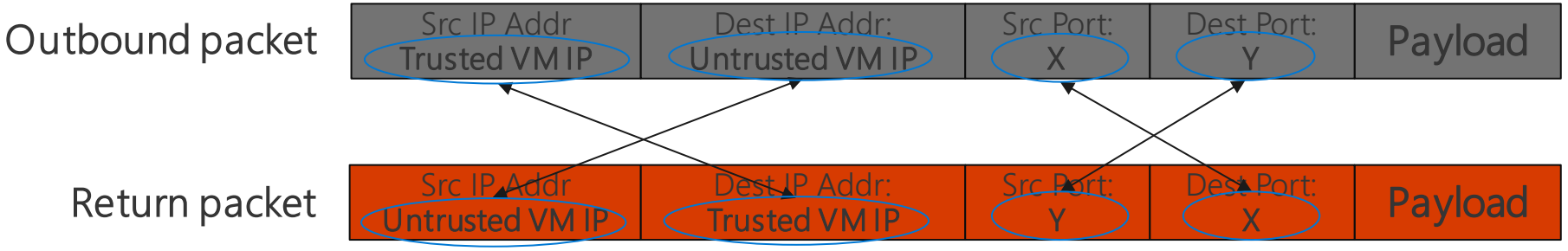
Flow symmetry with single NIC configuration
Why is it not an issue?



Outbound packet	Src IP Addr Trusted VM IP	Dest IP Addr: Untrusted VM IP	Src Port: X	Dest Port: Y	Payload
Return packet	Src IP Addr Untrusted VM IP	Dest IP Addr: Trusted VM IP	Src Port: Y	Dest Port: X	Payload

Flow symmetry with single NIC configuration
Why is it not an issue?

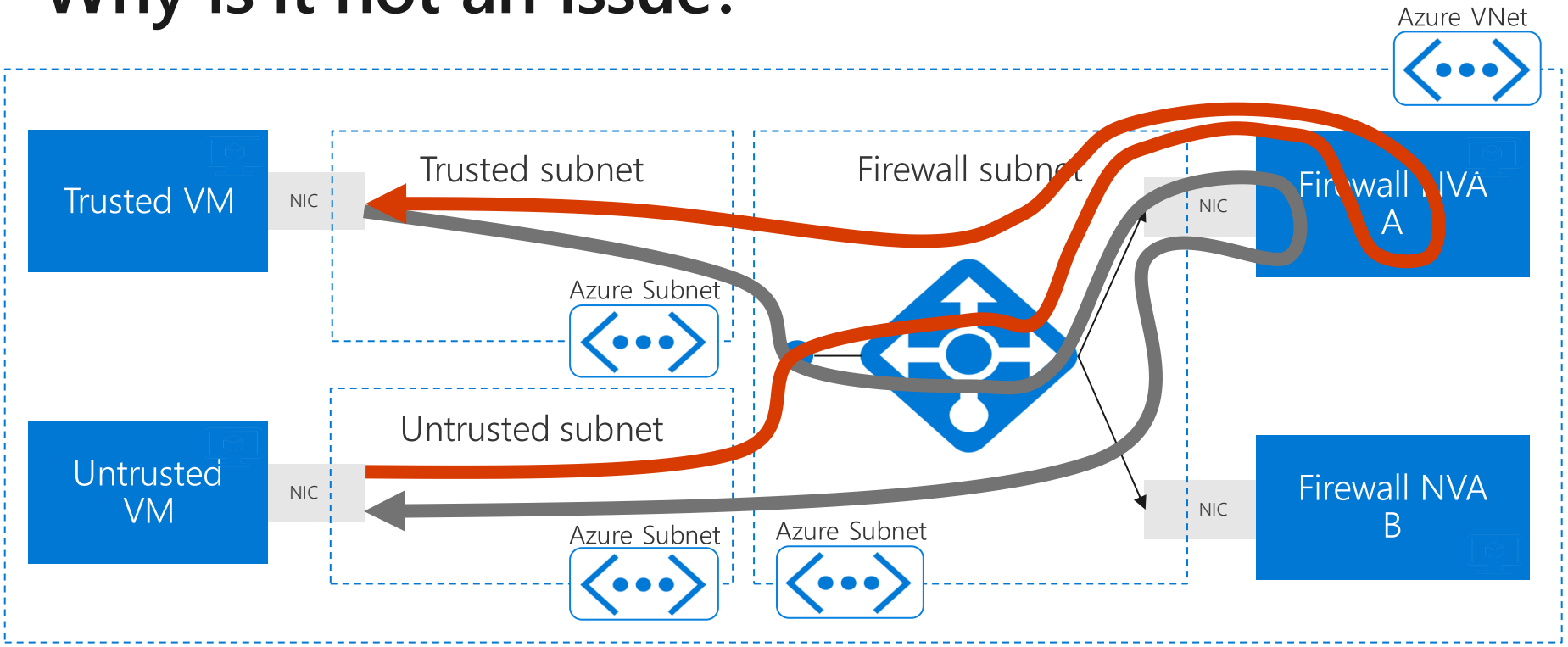




Both packets have the same src/dest IP addresses and ports, in reverse order

The load balancer's hashing algorithm assigns both packets to the same backend instance

Flow symmetry with single NIC configuration
Why is it not an issue?



Outbound packet	Src IP Addr Trusted VM IP	Dest IP Addr: Untrusted VM IP	Src Port: X	Dest Port: Y	Payload
Return packet	Src IP Addr Untrusted VM IP	Dest IP Addr: Trusted VM IP	Src Port: Y	Dest Port: X	Payload

You obtain flow symmetry!

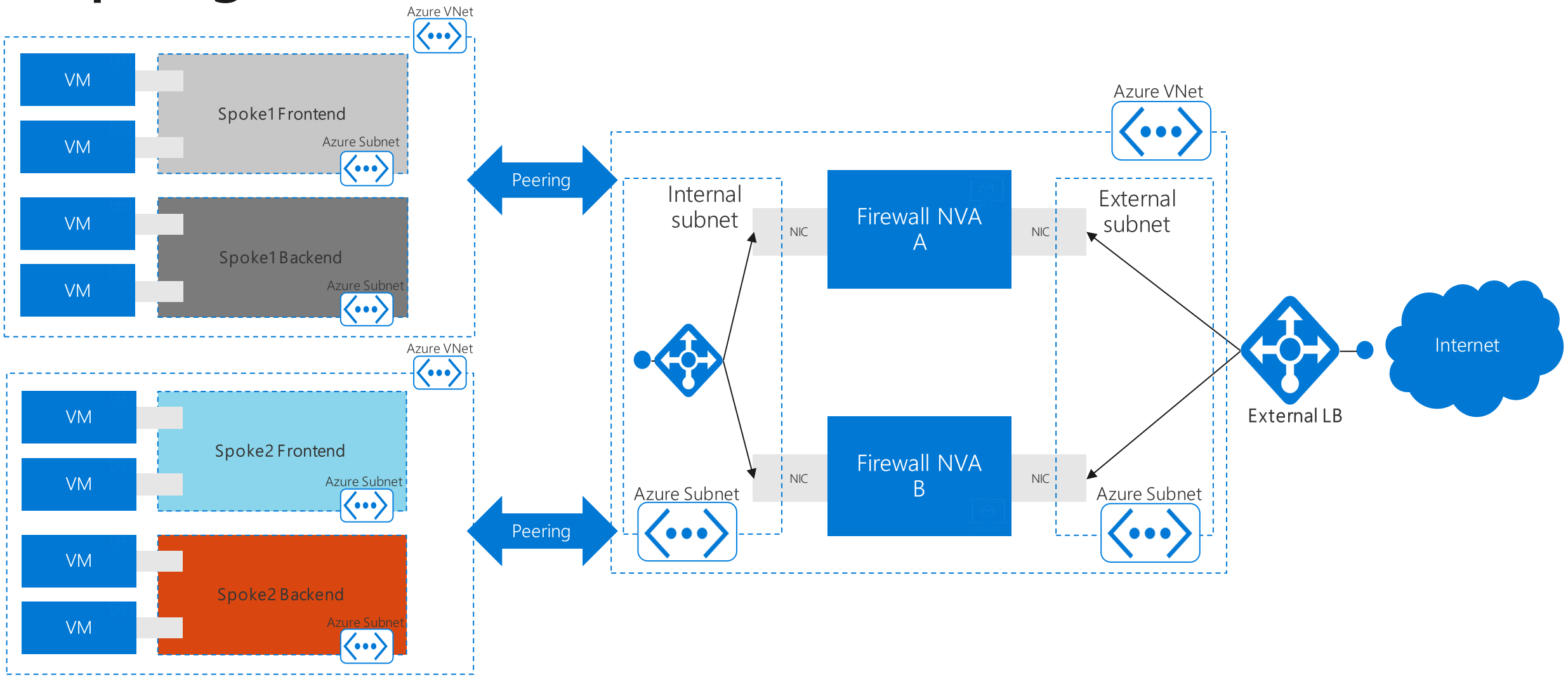
HA ports play a critical role here!

Return packet goes to port 56123, a randomly selected by the initiator of the connection

Active/Active with load balancer Vendor support

Vendor	Product (marketplace link)	Single NIC/Dual NIC	Documentation Available?	Notes
Fortinet	Fortigate Next-Generation Firewall (link)	Single, Dual, Multiple NIC	Yes (link, Github templates)	
Checkpoint				Work in progress

Edge security + segmentation for hub & spoke topologies



Edge security + internal segmentation

Session Takeaway

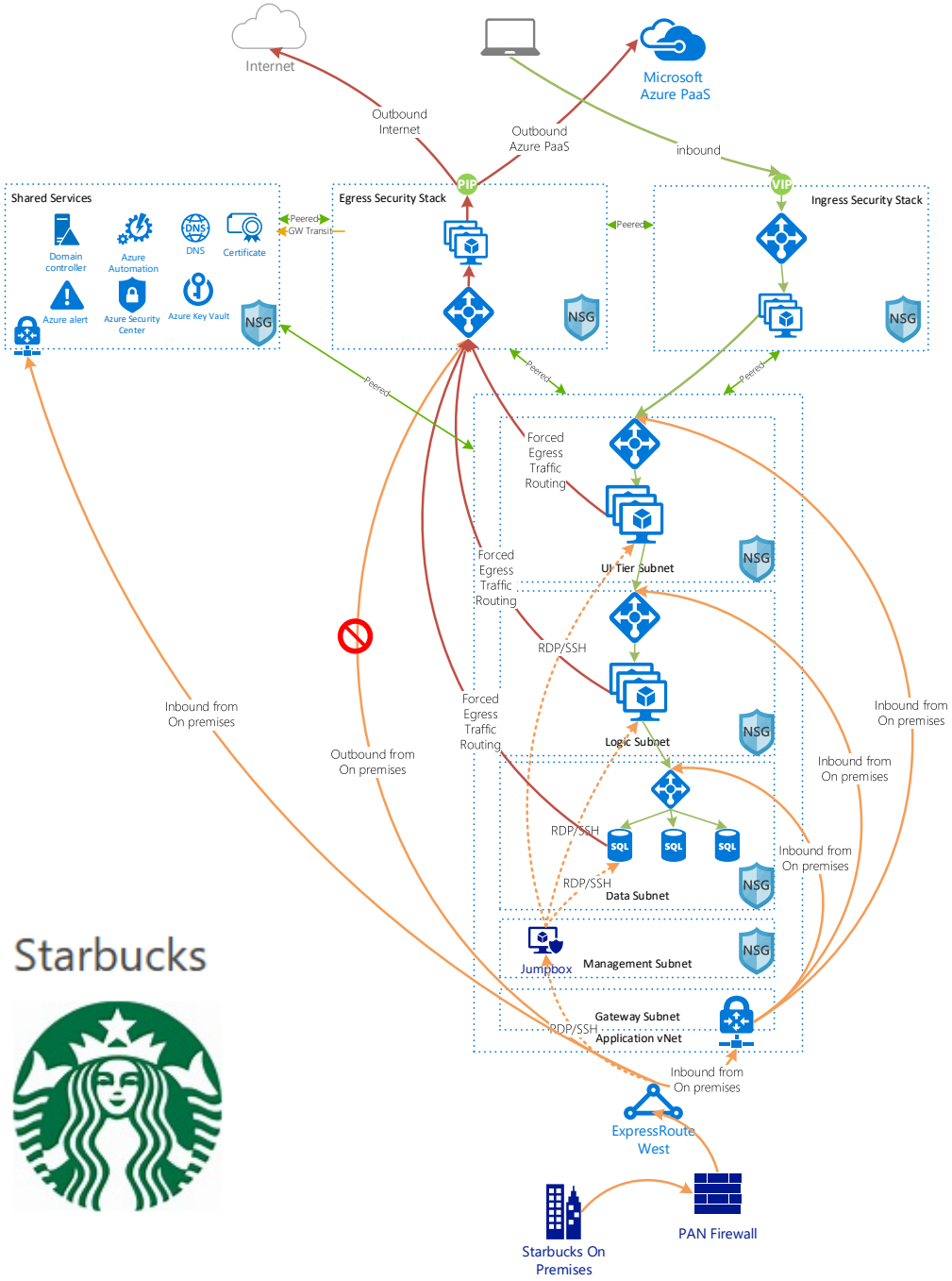
Same NVA cluster used for edge security and internal segmentation

NAT for inbound internet traffic cannot be avoided

NAT not required for internal traffic, with single NIC configuration

Can be extended to hub & spoke topologies

Recommended architecture for DMZ



Example customer virtual network

Centralized Security Stacks

Centralized Shared Services

Virtual Network Peering:

- Within region
- Across subscriptions
- Gateway Transit

User Defined Routing to force traffic out the Egress stack

NSGs for micro-segmentation within application tiers and between applications

ExpressRoute circuit per region

Microsoft

© Copyright Microsoft Corporation. All rights reserved.

Azure Virtual Network Reference

Routing Example

ARM templates

Services with Vnet Integration

Accelerated Networking

DNS in Vnet

UDR example

Azure Network Topology Reference

VPNGateway Transit Example

Hub-Spoke Example

Azure DMZ & NVA

On-Prem ER with Failover VPN

Vnet Peering Example

Network Watcher Example

Vnet FAQ

Azure NSG, DDoS and Service Endpoint Reference

NSG Tutorial

NSG Troubleshoot

NSG Logs

Simple DMZ with NSGs

Azure DDoS Configuration

Azure Storage & ServiceEndpoint

Azure SQLDB & ServiceEndpoint