UK IPv6 Council

IPv6 Cloud Workshop

IPv6 in Azure

Fabrizio Ferri

Azure Program Manager

fabrizio.ferri@microsoft.com

3-Sept-2019

Why IPv6?

Lot's of IPv6 detail on Wikipedia:
https://wikipedia.org/wiki/IPv6

The World is OUT of new IPv4 address blocks

AFRINIC (Africa's Regional Internet Registry) deployed the last large (/8) IPv4 range in Nov2018

- PRIMARY IPv6 Benefit: Enormous Address Space – 3.4×10^{38} addresses
  Think: Mobile Devices, Cars, Light Bulbs (IoT), etc.

- Other IPv6 Benefits:
  - Simplified and extendable header allows for faster processing by routers
  - Large unique address pool allows for less network address translation (NAT) which simplifies large networks and increases performance
  - Both stateful and stateless auto configuration- thus absence of a DHCP server does not block communication.
  - Native security- IPSec is built into the IPv6 protocol
  - Built for a mobile world – IPv6 clients can roam across geographical areas while remaining connected with same IP address. IPv6 mobility leverages auto IP configuration and extension headers.

What is IPv6 for Azure Virtual Network?

Preview

- IPv6 VNET Preview lunched on 23 Apr 2019 in ALL Azure Public Regions

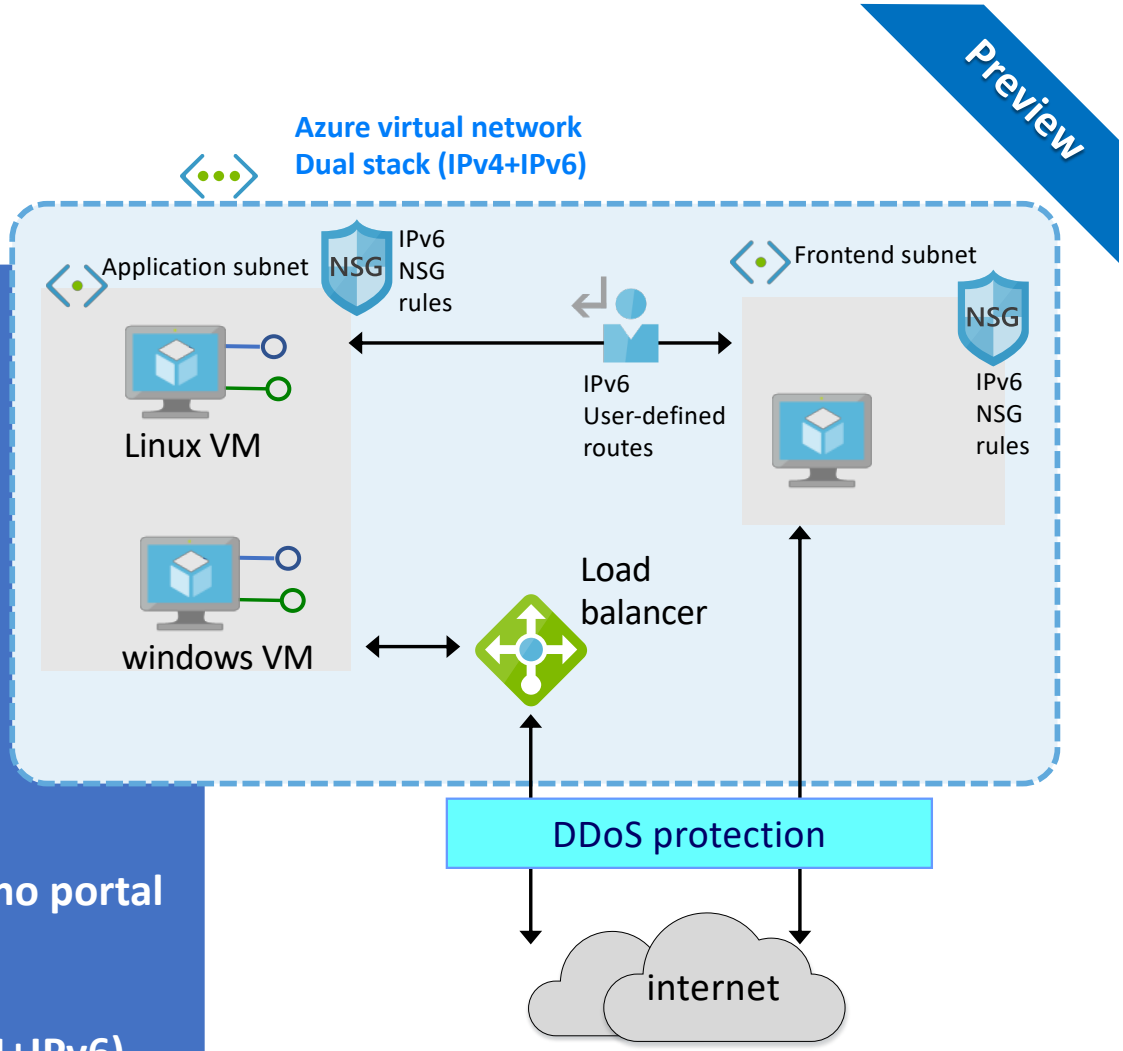
https://azure.microsoft.com/en-us/updates/public-preview-microsoft-adds-full-ipv6-support-for-azure-vnets/

- host applications in Azure VM with IPv6 and IPv4 connectivity to extend applications to Mobile (phone) and IoT markets
- VMs in Azure with Dual stack IPv4/IPv6 connectivity:
  - inside the VNet
  - from/to internet
- Supports both Windows & Linux VMs
- Advantage of dual stack: flexibility in deployment
- Customer-defined IPv6 address space for ease cloud migration and integration with on-premises

IPv6 in Azure VNET

IPv6 in Azure VNET Feature Set

- **Customer-defined IPv6 space in VNET**
- **IPv6 Network Security Group (NSG) rules**
- **IPv6 User-Defined Routes (UDR)**
- **Support Win+Linux VMs**
- **Instance-Level Public IPv6**
  (*IPv6 Internet connectivity directly to individual VM*)
- **Basic Load Balancer**
- **Standard Load Balancer w/ IPv6 Probe**
- **Azure DNS**
- **VM Scale Set**
- **Portal- View-only**
- **Network Watcher: NSG flow logs, packet capture (no portal filters)**
- **GetEffectiveRules & GetEffectiveRoutes**
- **Azure portal: create/edit/delete of dual stack (IPv4+IPv6) VNets and subnets, IPv6 NSGs, IPv6 UDRs, IPv6 public IPs.**

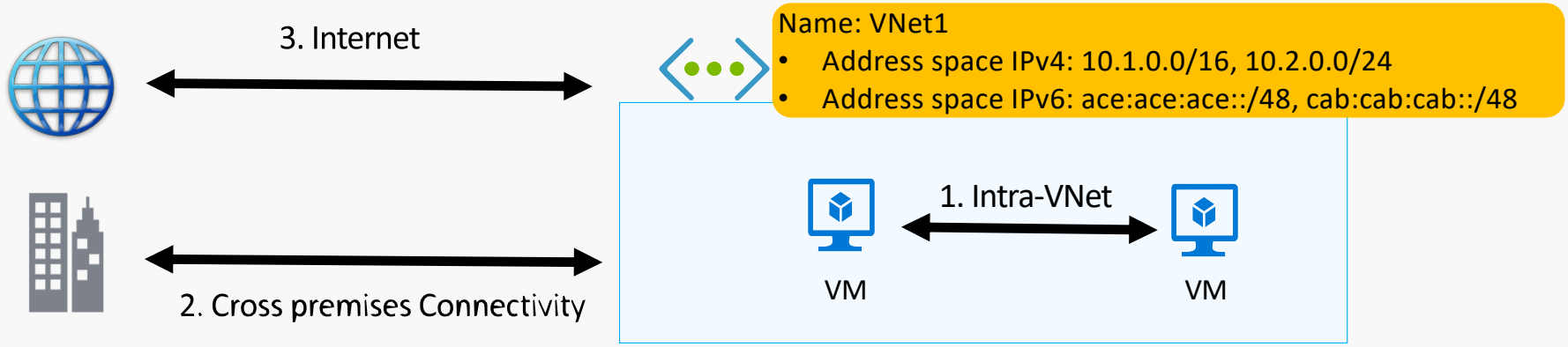


Virtual Network

Isolated, logical network that provides connectivity for Azure VMs

User-defined address space (can be one or more IP ranges, not necessarily RFC1918)

1. Connectivity for VMs in the same VNET
2. Connectivity to external networks/on-prem DC's
3. Internet connectivity

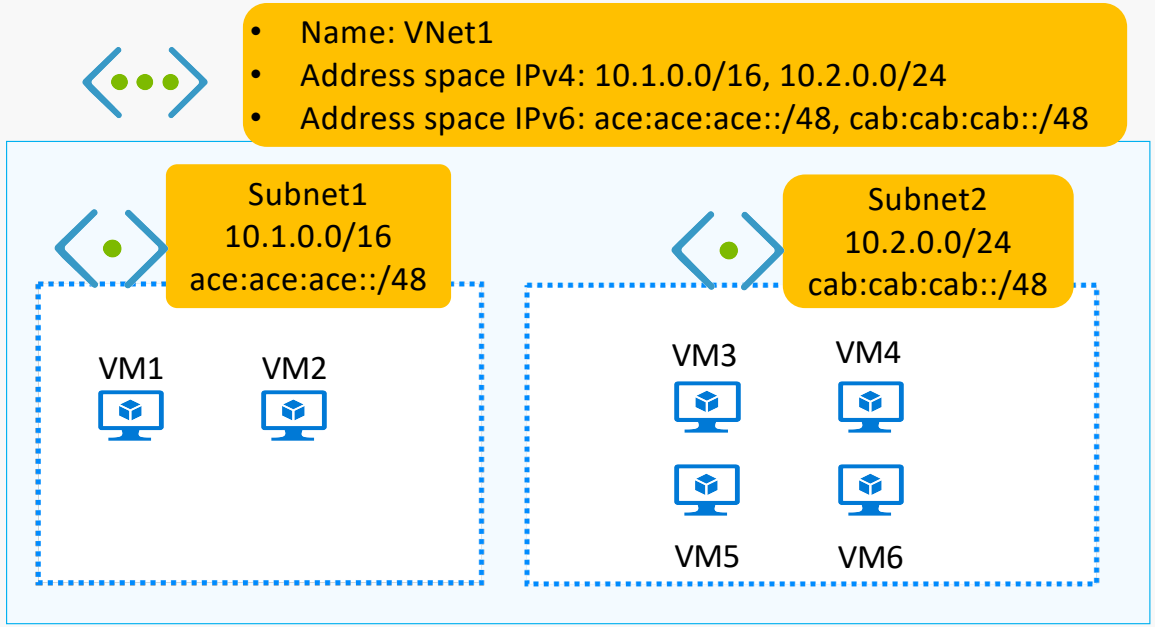


Subnet

IP subnet

- Provides full layer-3 semantics and partial layer-2 semantics (DHCP, ARP, no broadcast/multicast)
- Subnets can span **only one range of contiguous IP addresses**

VMs can be deployed only to subnets (not VNETs)



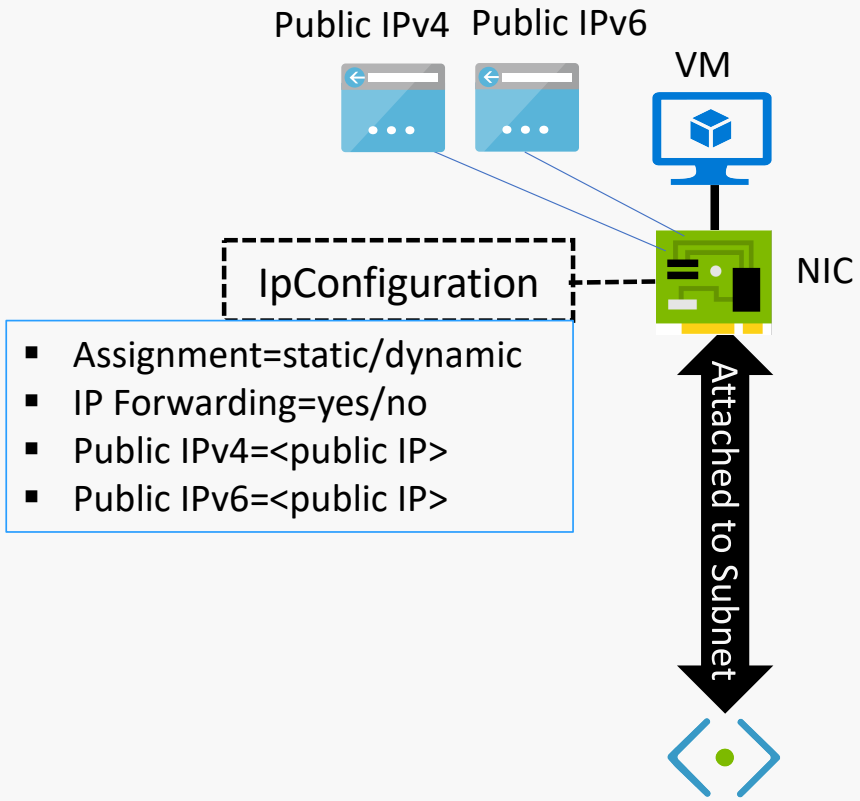
- in Azure VNet, packets can flow between to different subnets without explicitly traversing any layer-3 device.
- Traffic filtering can be applied by NSG

Network Interface

Virtual NIC that connects a VM to a Subnet

- One private IP address (private == included in the subnet's IP range)
- Private IP address always assigned via Azure DHCP

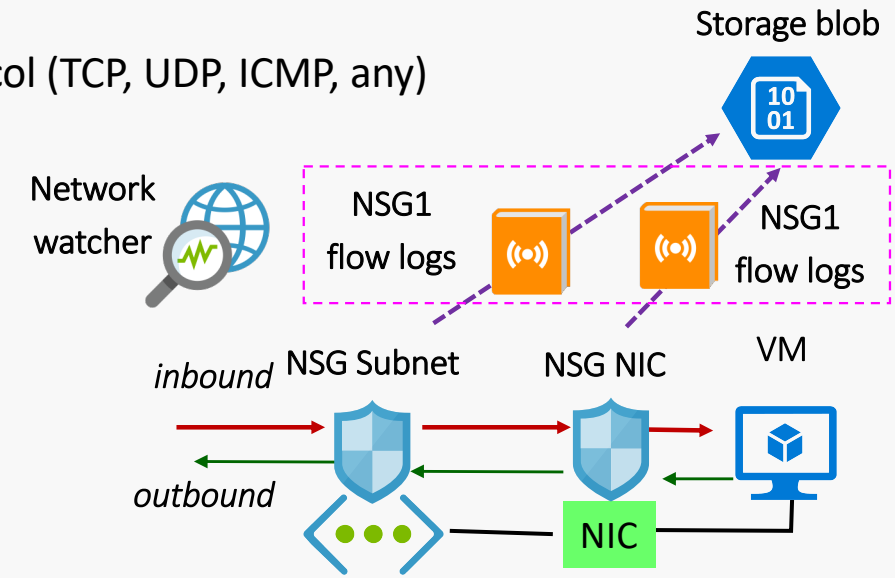
  - *Dynamic assignment* = DHCP assigns new IP when VM is re-allocated
  - Static assignment =DHCP assigns always the same IP
  - IP forwarding = NIC can receive/hand off packets with dest IP address different from its private IP
  - Public IP = NAT address associated to the NIC



NSG to control the traffic

5-tuple ACL's

- Source IP, Destination IP, Source Port, Destination Port, Protocol (TCP, UDP, ICMP, any)
- Actions: allow or deny
- Directions: inbound, outbound
- Priority: 100-4096 (lower value = higher priority)



Stateful

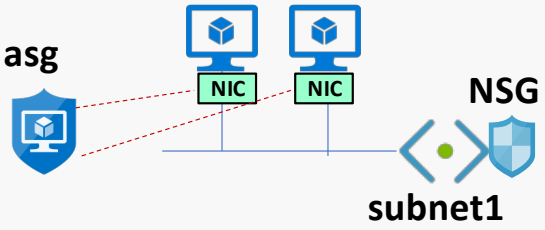
No need to define rules for «return traffic»

Can be applied to NICs and Subnets (ARM)

- Inbound connections: subnet-level NSG evaluated first, NIC-level NSG evaluated next
- Outbound connections: NIC-level NSG evaluated first, subnet-level NSG evaluated next

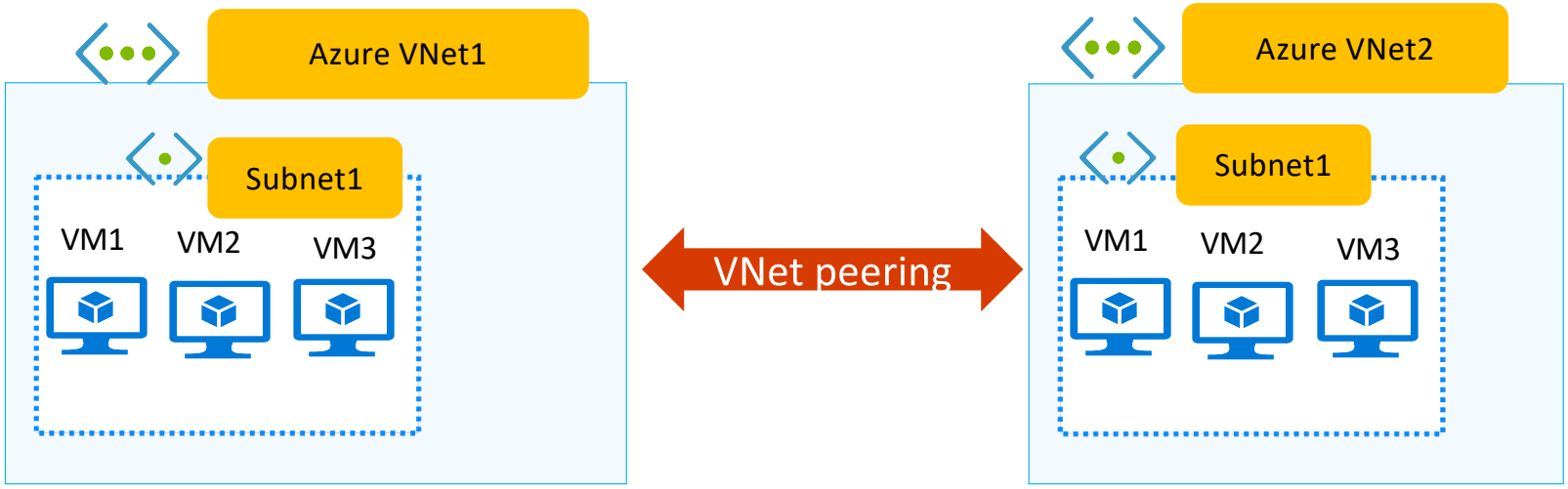
Application security group

- Provides the capability to group VMs with monikers, instead of explicit IP addresses
- network security policies (allow/deny) based on workloads

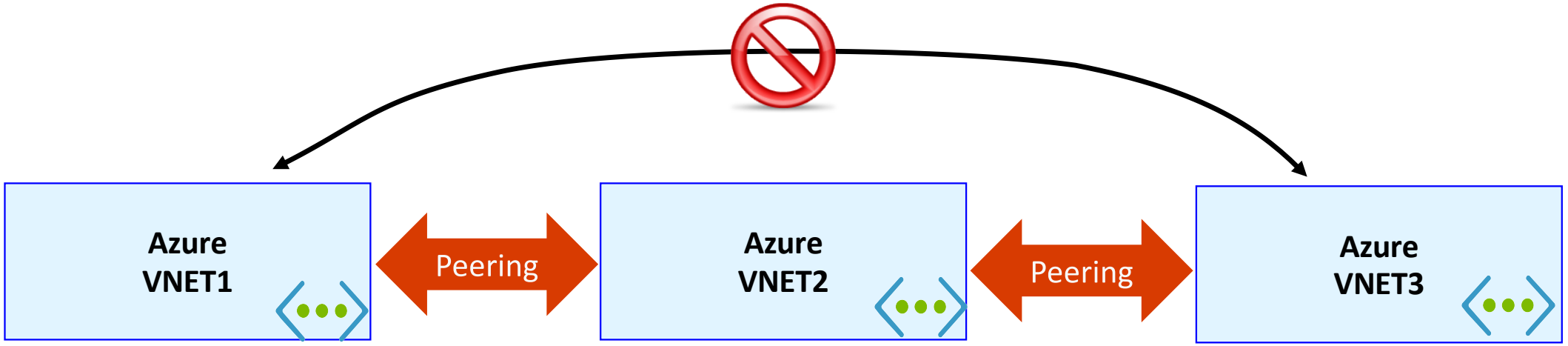


What is VNet peering?

- Ability to “merge” two Azure VNets, so that VMs in the two VNets can communicate with each other as if they were on the same VNet
- VNets can be in same or different Azure regions
- VNet peering traffic routed through the Microsoft Backbone (no public internet involved)
- Allow direct VM-to-VM connectivity, no extra hops



VNet peering is non-transitive



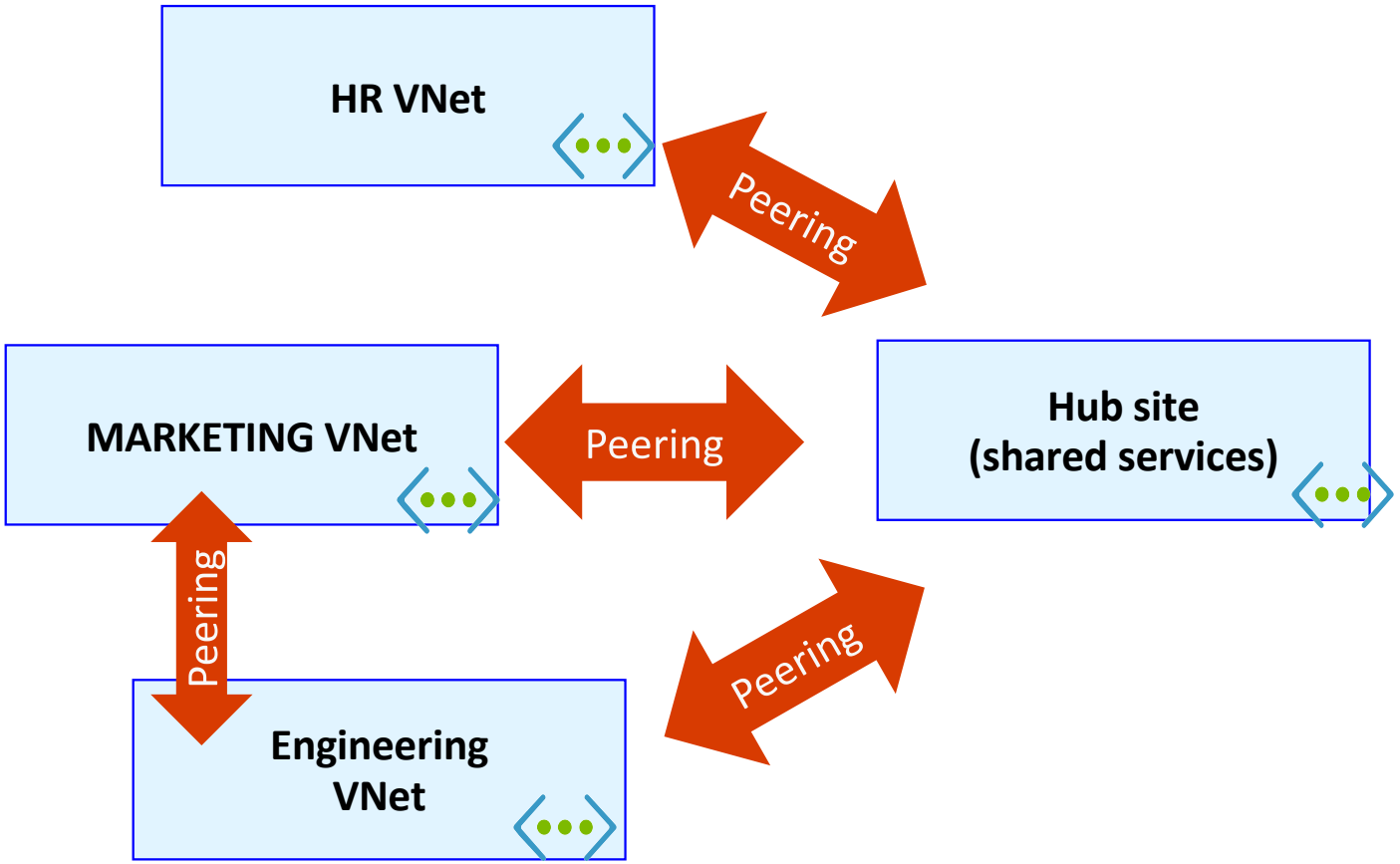
VNet1 can send/receive traffic to/from VNet2

VNet2 can send/receive traffic to/from VNet1, VNet3

VNet3 can send/receive traffic to/from VNet2

VNet1 cannot send/receive traffic to/from VNet3

VNet peering topology:Hub&Spoke, partially/full mesh



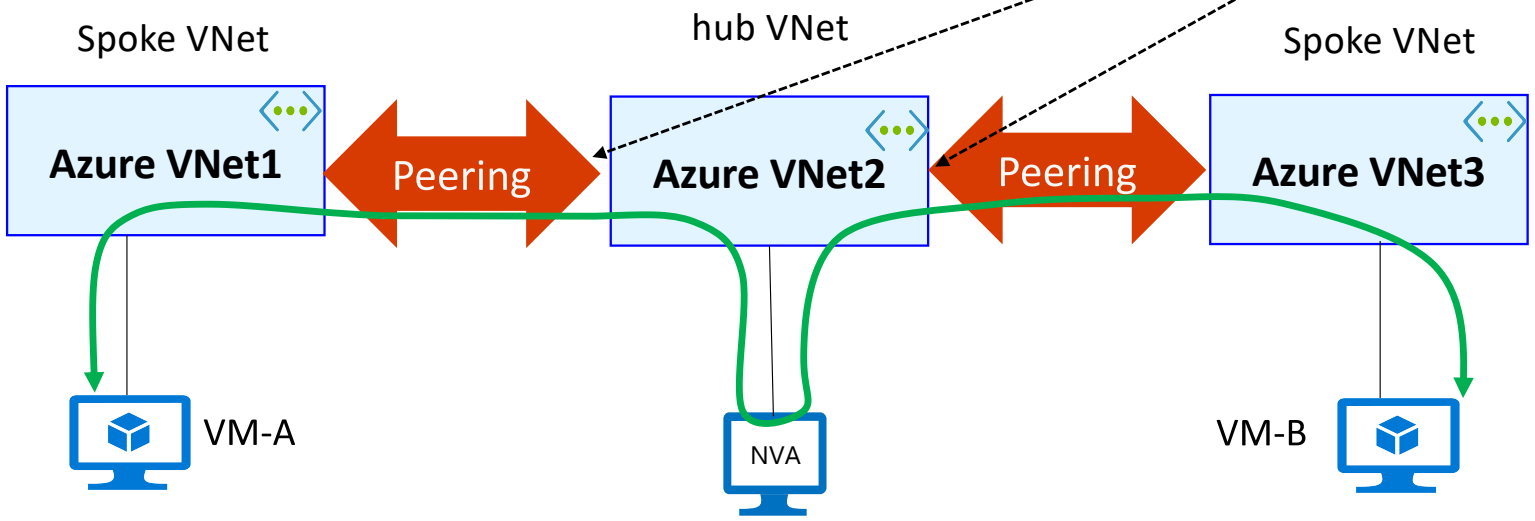
All VNets can communicate with the Hub VNet,

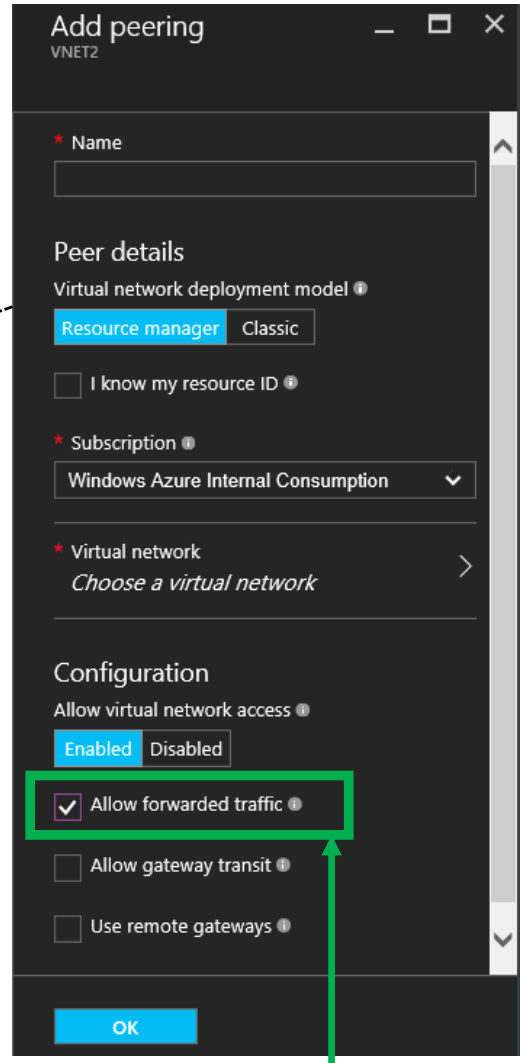
but

HR, Marketing and Engineering cannot “talk” to each other

Transit routing in VNet peering

- A VNet's ability to route traffic which has not been originated in it and is not destined to it



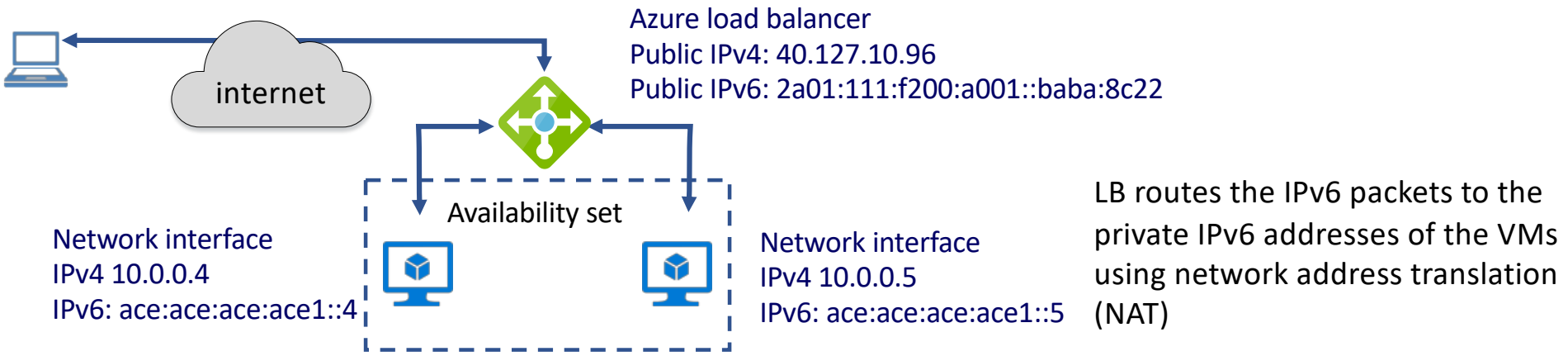


It is necessary when traffic originated in the peered VNet (VNet1) need to be forwarded to a destination VNet (VNet3) by transit in middle VNet (VNet2)-so that *transit* traffic can route through the hub VNet (VNet2)

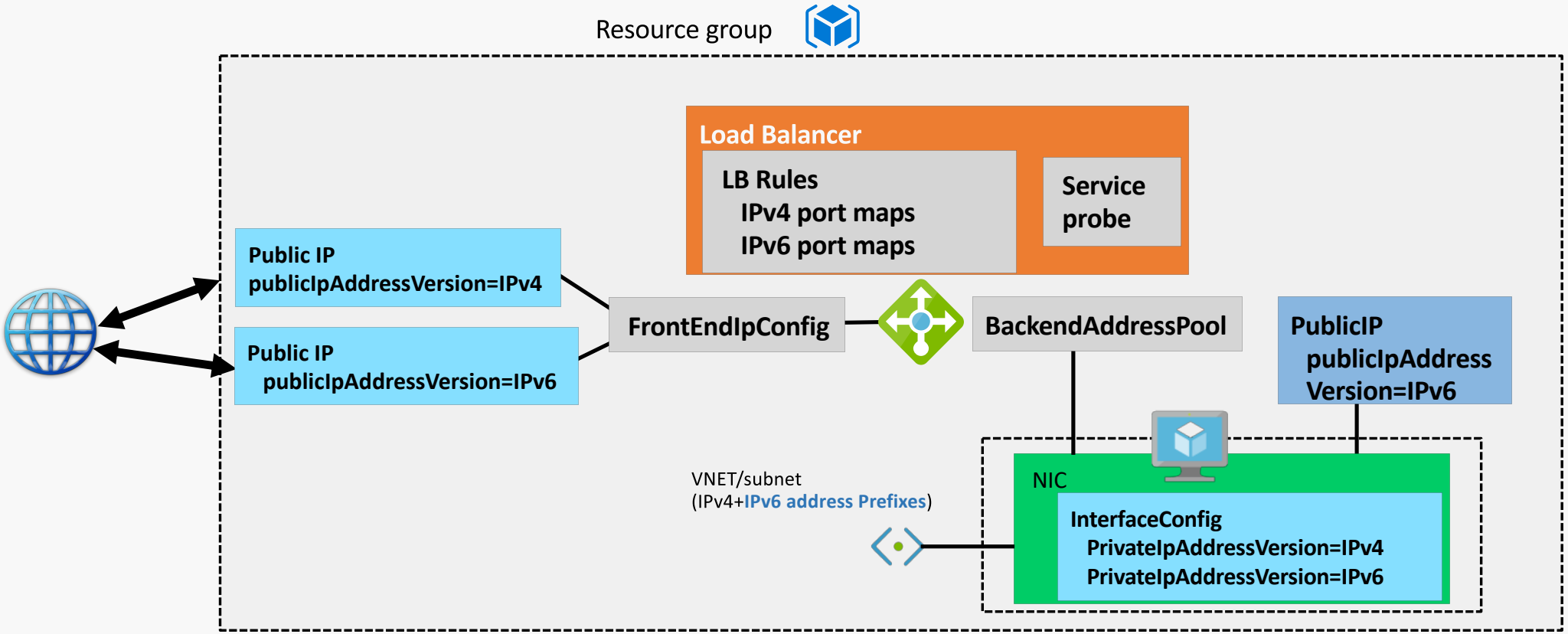
It can be allowed/denied

IPv6 LB

- Internet-facing load balancers can be deployed with an IPv6 address
- IPv6 **inbound connectivity:** Internet IPv6 hosts -> Azure VMs
- IPv6 **outbound connectivity:** Azure VMs -> Internet IPv6 hosts
- LB routes the internet IPv6 packets -> VM private IPv6 addresses using NAT



IPv6 LB: config details



User Defined Routes

Additional static routes that modify a VNET's default routing policy

- UDR can be applied to subnet
- UDR acts on traffic leaving NICs, based on destination address

3-tuples

Address Prefix: range of destination IP addresses
Next hop type: "**Virtual Network**", "**Virtual Network gateway**", "**Virtual Appliance**", "**None**"
Next hop IP address: only when next hop type is "Virtual Appliance"

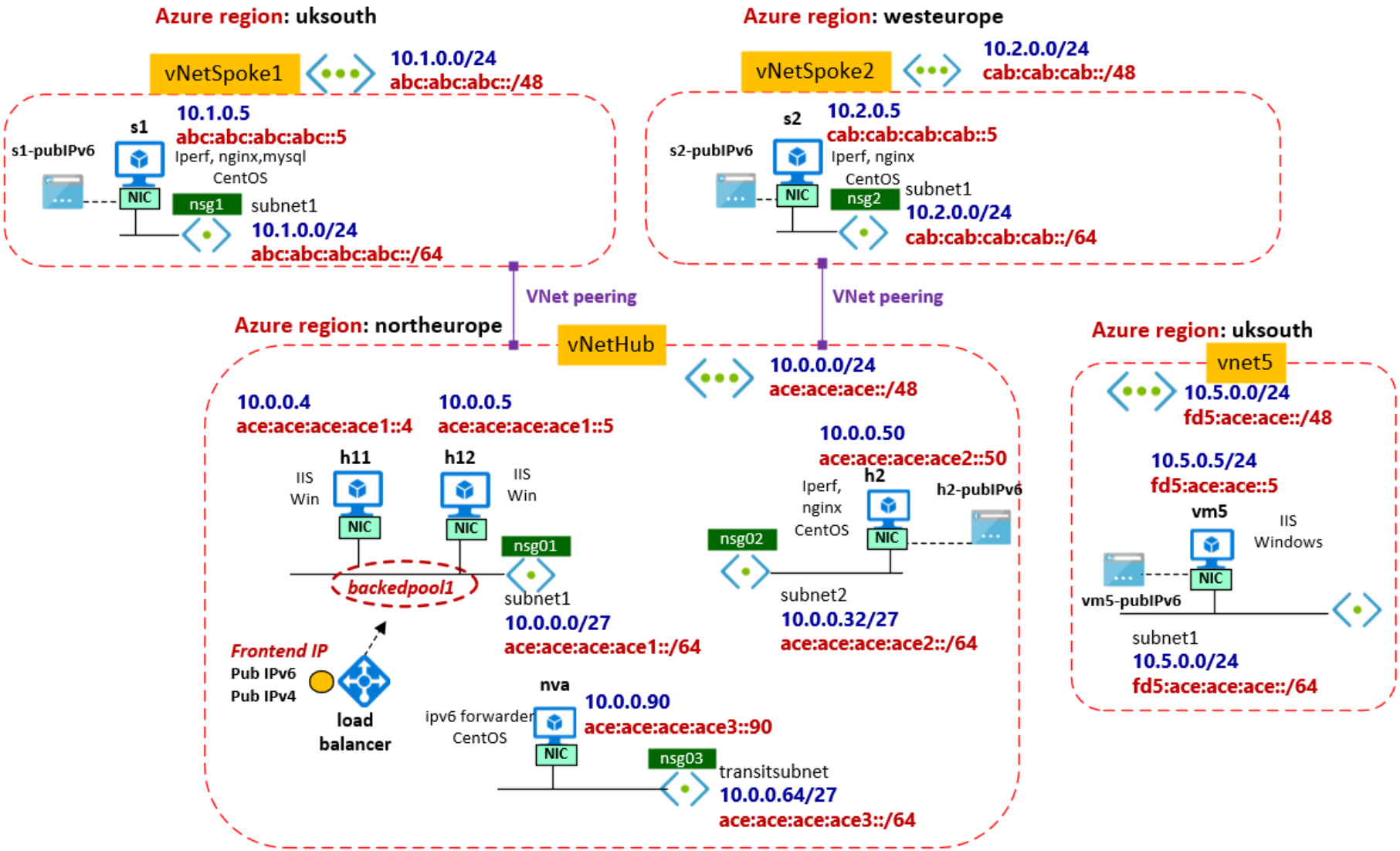
black hole

Longest Prefix Match (LPM)

A packet is routed according to the UDR rule that best matches its destination address
Best match = longest prefix match
If a packet matches two or more rules with the same prefix length, weight is used::
Priority: 1. user-defined routes; 2.BGP routes; 3. System routes

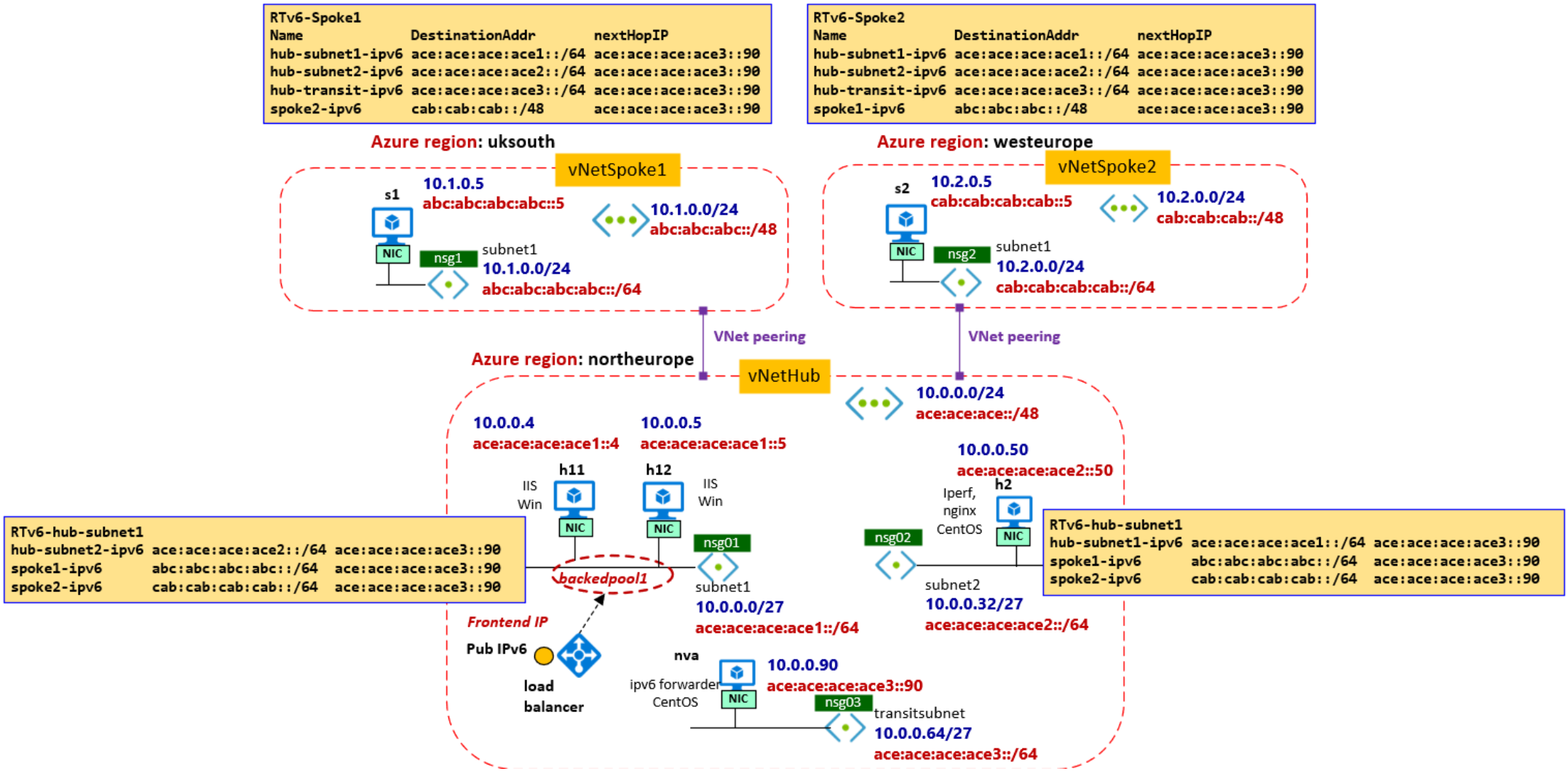
IPv6 hub-spoke VNet in peering

https://github.com/fabferri/az-pattern/tree/master/01-ipv6-vnet-peering



UDR

https://github.com/fabferri/az-pattern/tree/master/01-ipv6-vnet-peering



Q&A

Reference

https://docs.microsoft.com/en-us/azure/virtual-network/ipv6-overview

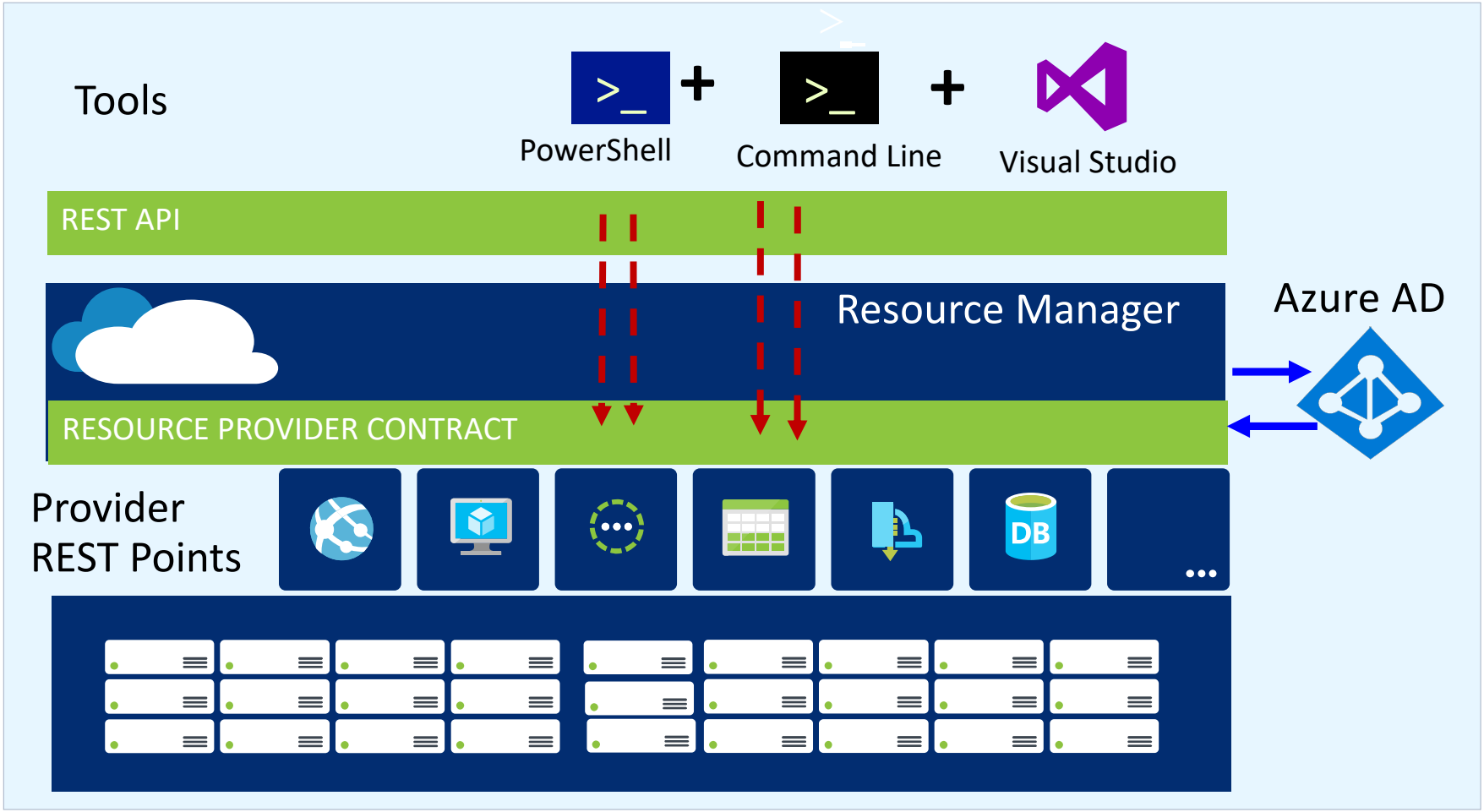
https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-ipv6-overview

https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-ipv4-ipv6-dual-stack-powershell

https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-ipv4-ipv6-dual-stack-standard-load-balancer-powershell

Introducing Resource Manager

Management Layer



How to deploy resources in Azure: imperative vs declarative

imperative

Commandlets

```
New-AzVM –VM $myVM
New-AzStorageAccount –Name $acct
New-AzVirtualNetwork –Name $vnetName
```

Text file with JSON syntax

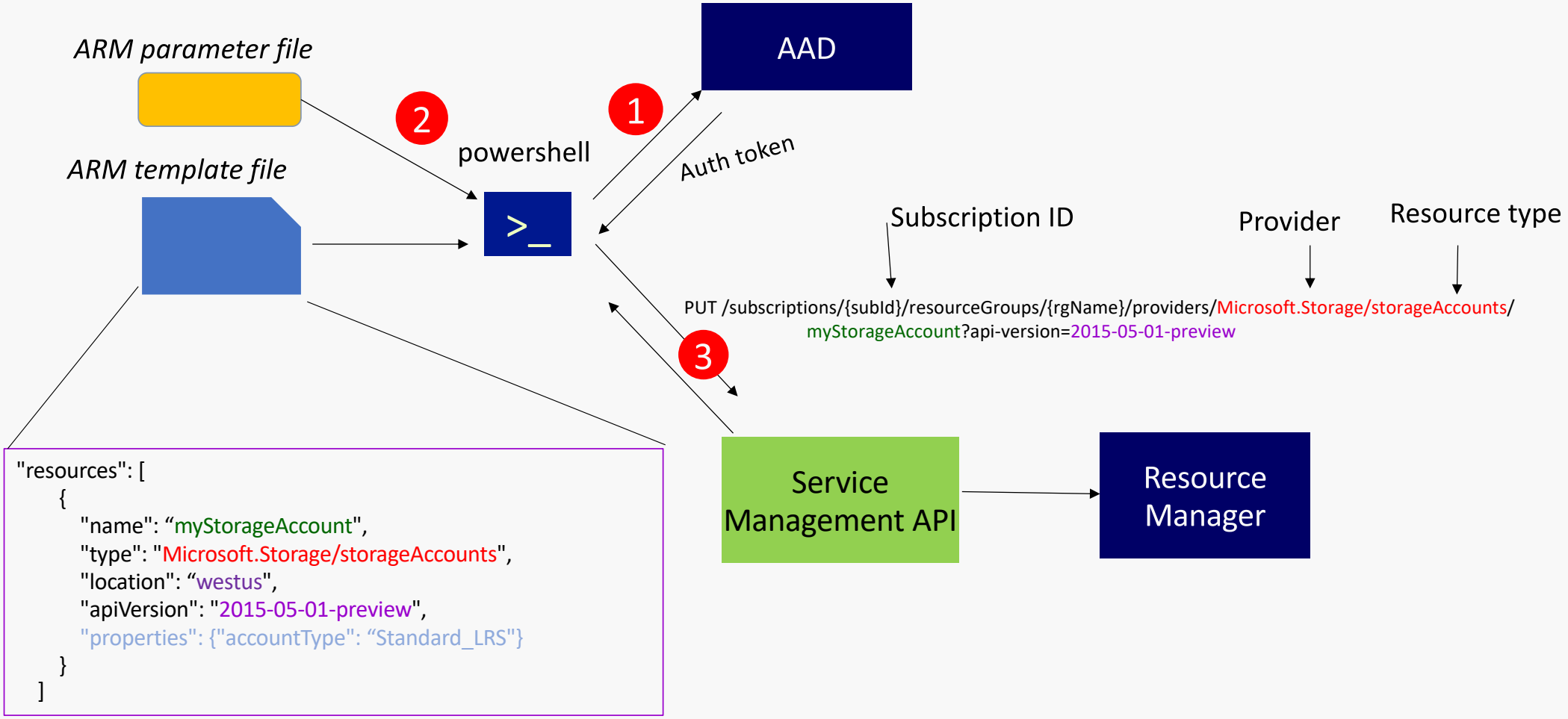
declarative

Template

```
{
 "$schema": "https://../deploymentTemplate.json#",
 "contentVersion": "1.0.0.0",
 "parameters": {},
 "variables": {},
 "resources": [],
 "outputs": {}
}
```

You decide

Deployment by ARM template



Model Based Deployment

Power of Repeatability

ARM Template: ***declarative JSON file*** that defines the goal state of a deployment

An ARM Template is a:

- Source file, checked-in
- Specifies resources and dependencies
- Parametrized input/output

```
"parameters": {
 "sharedKey": { "value": "myshare"
}
```

```
"parameters": {
 "sharedKey": { "type": "string"}
}
```

ARM Template can:

- Ensure Idempotent
- Simple Orchestration
- Cross-Resource Configuration & Update