Microsoft

Azure Activate for App Service Environment

Architecture

Conditions and terms of use

1 For a more detailed description of confidentiality, see the slide at the end of the presentation.

Learning Units covered in this Module

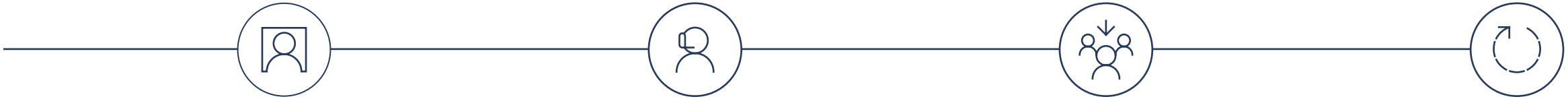
- App Service Architure
- App Service Environment Architecure
- Differences between App Service Vs ASE
- ASEv2 vs ASEv3

Objectives

After completing this Learning, you will be able to:

Understand….

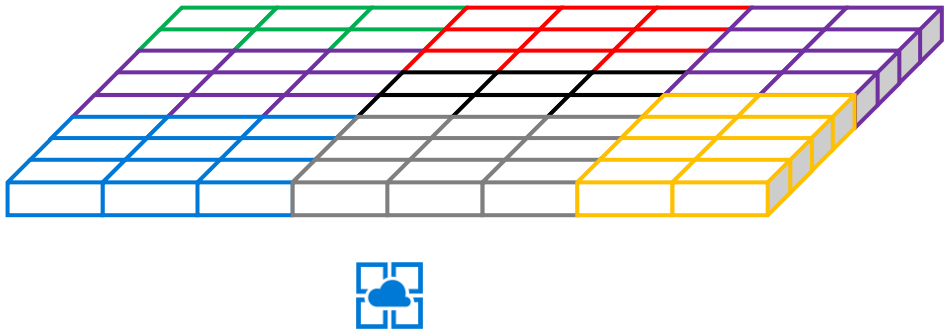
- ✓ Explain App Service Architecture
- ✓ Explain App Service Environment Architecture
- ✓ Explain Differences between multi-tenant App Service vs ASE
- ✓ Explain ASEv2 vs ASEv3



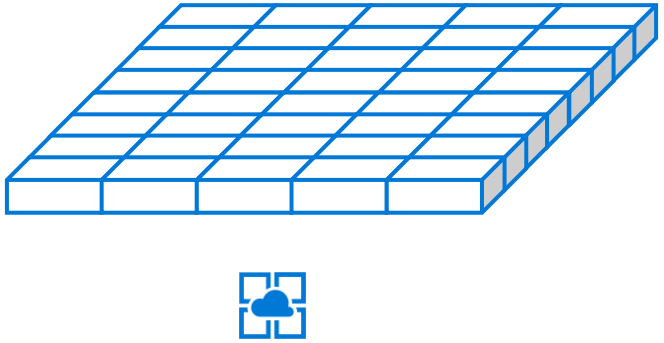
Networking

App Service and App Service Environment

The App Service is a multi-tenant service. Thousands of customers that share the same system safely and securely.



An App Service Environment (ASE) is a single tenant instance of the App Service that deploys into the customer's Azure Virtual Network (VNet)

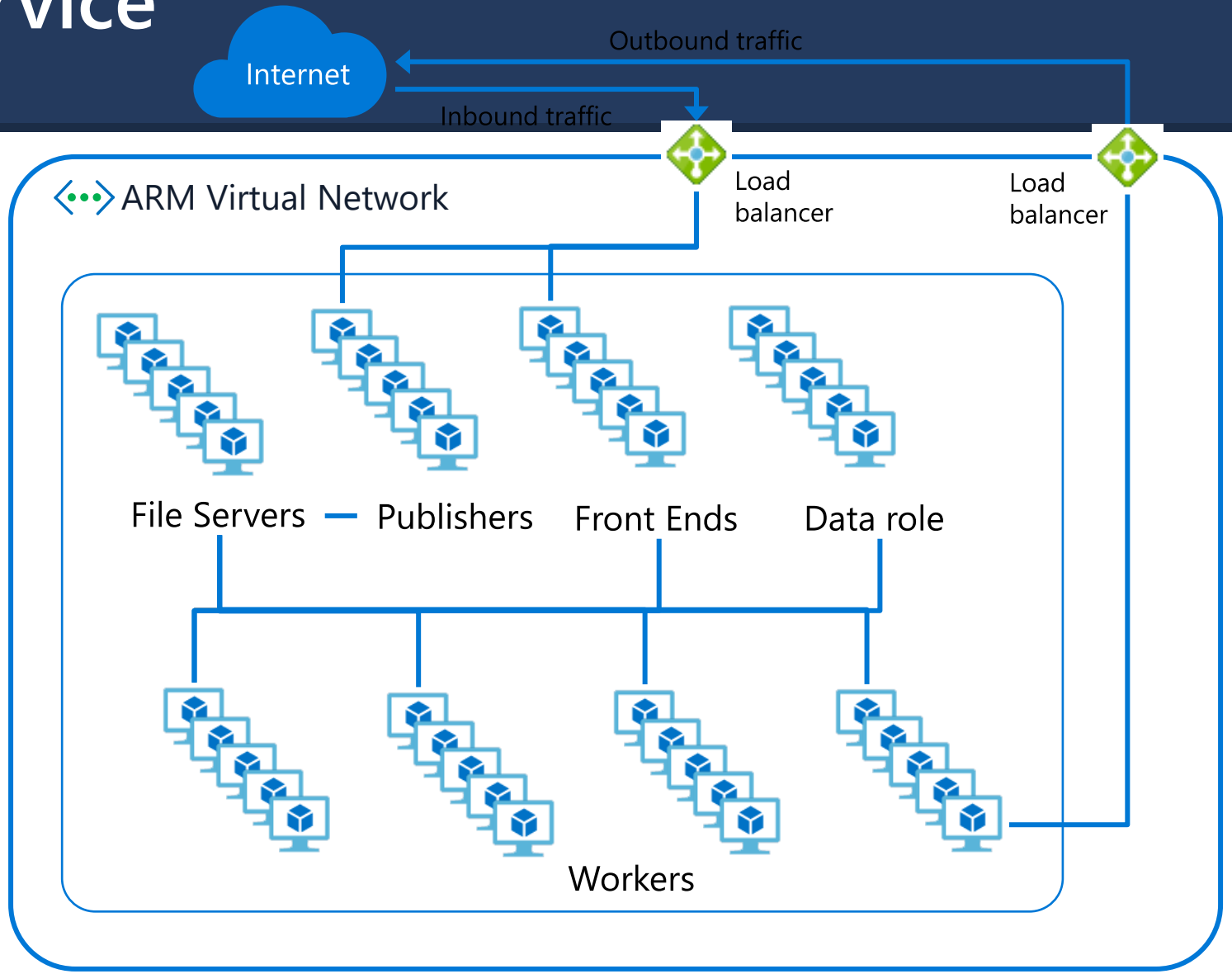


Multi-tenant App Service

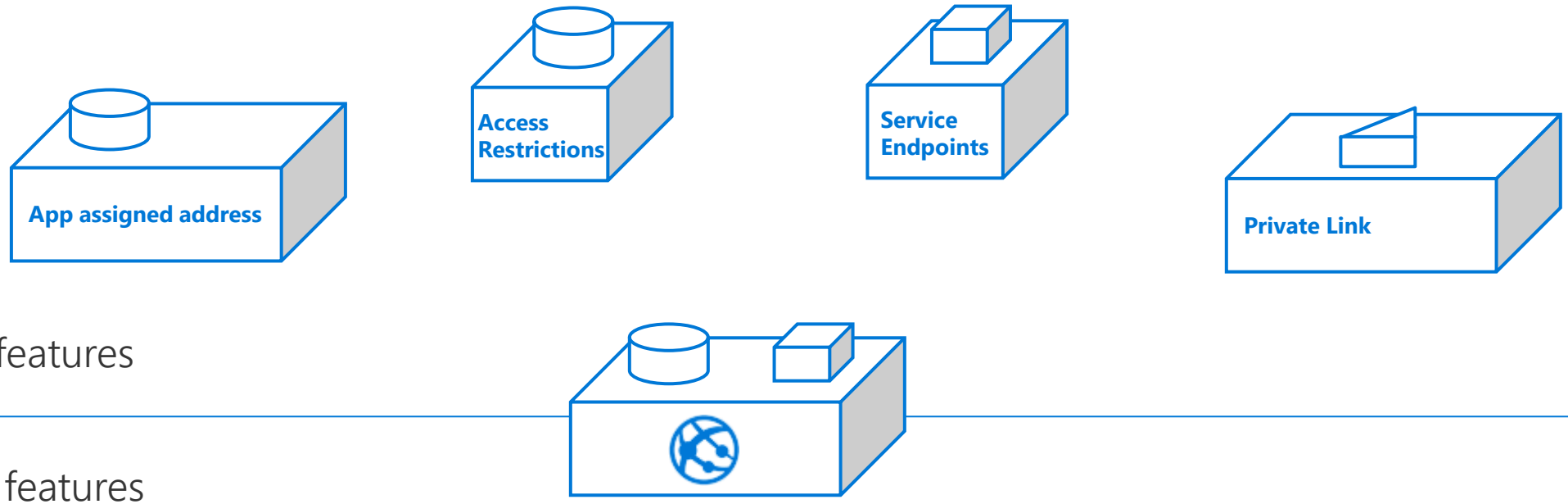
- VMSS based deployments
- Multi-tenant deployments with thousands of customer apps.
- Distributed platform with multiple roles.
- HTTP/S terminates at Front Ends.
- Customer workloads run on the workers.

Addresses:

- Inbound HTTP/S VIP: 1
- FTP address: 1
- Shared outbound VIPs: 36



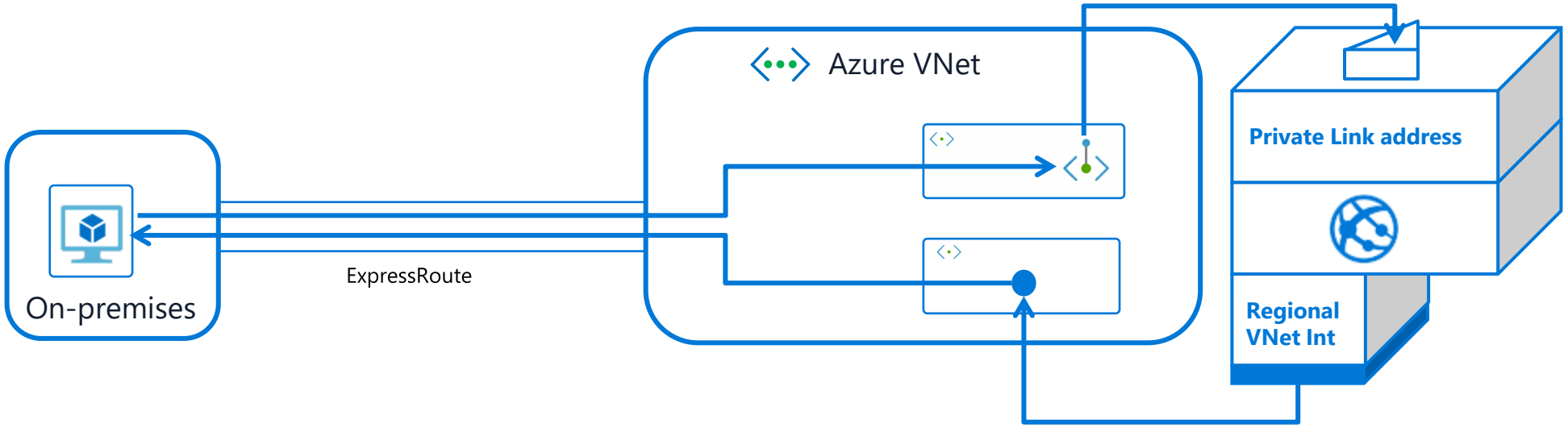
Multi-tenant App Service networking features



Inject an app in a VNet

Add a private endpoint to your app

Add VNet Integration to your app

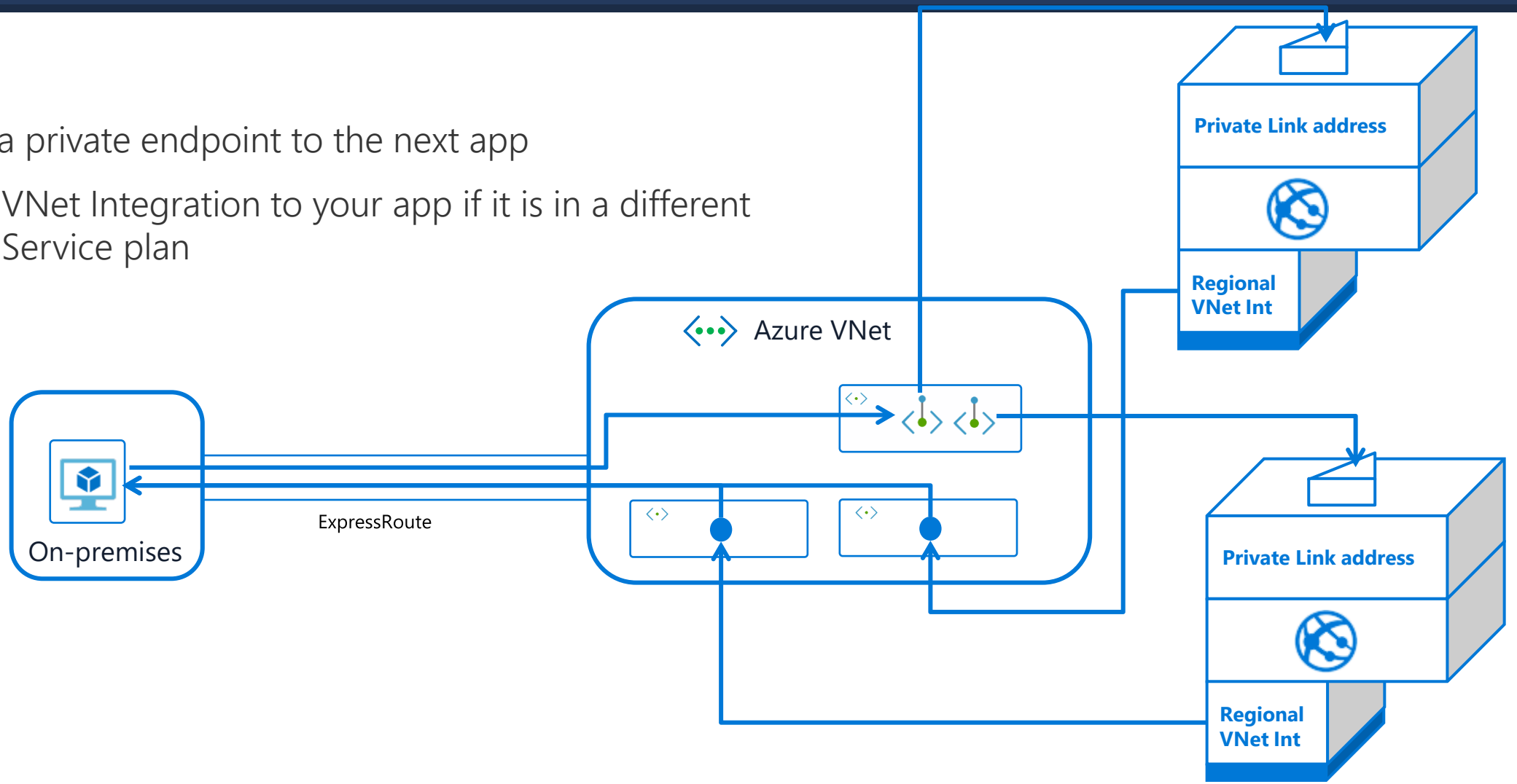


Use policies to audit that you have private endpoint and/or VNet Integration

Inject two apps in a VNet

Add a private endpoint to the next app

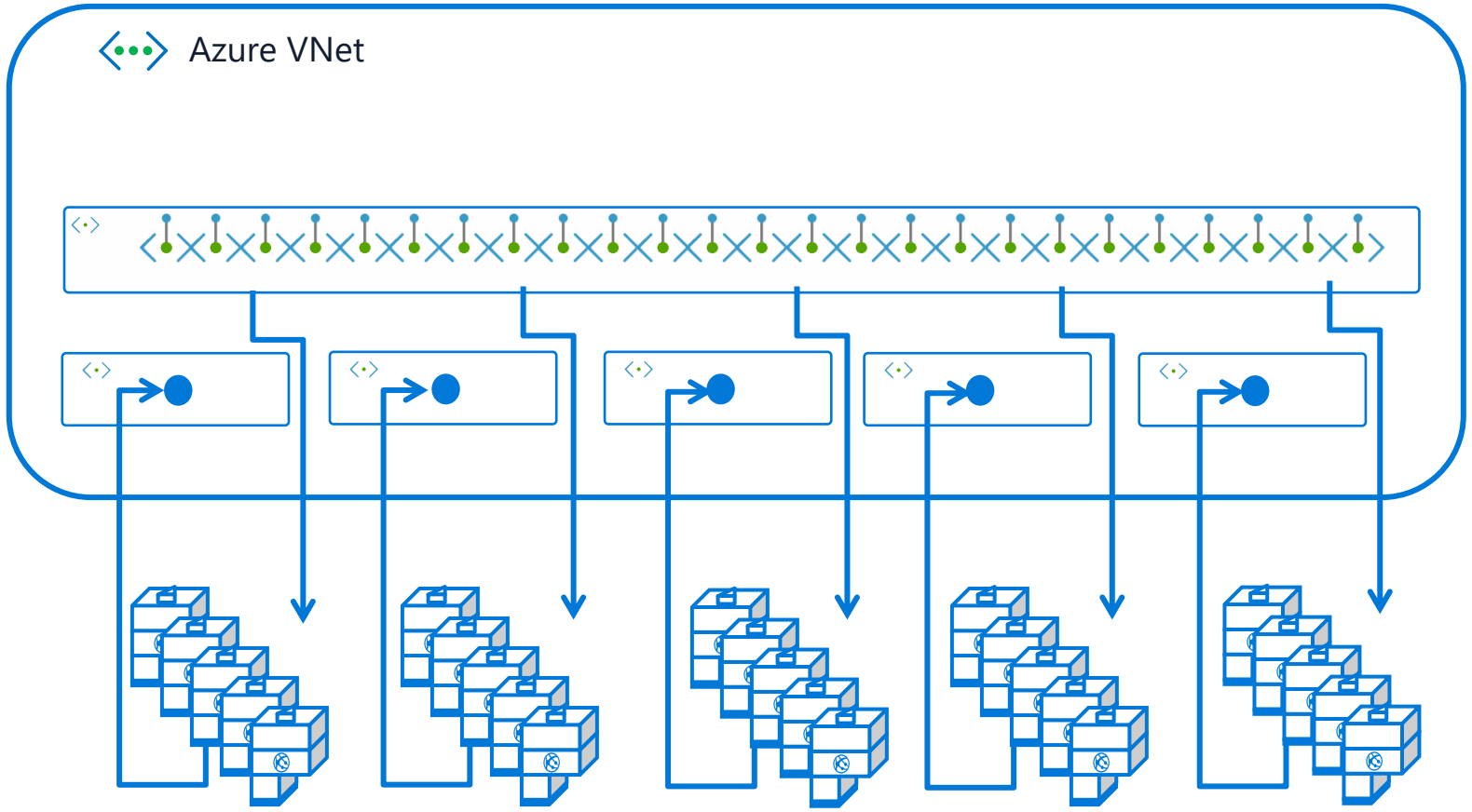
Add VNet Integration to your app if it is in a different App Service plan



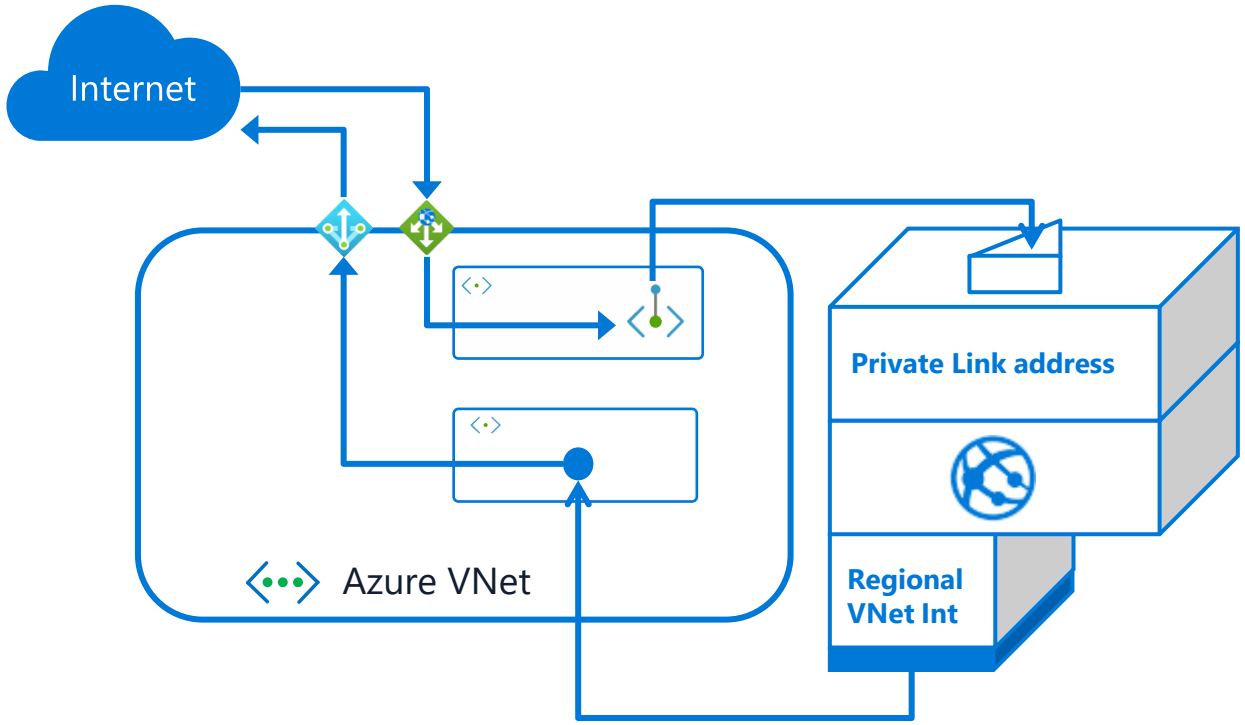
Injecting many apps

Injecting 25 apps in 5 App Service plans will use up 5 subnets and 25 private endpoints.

(One ASEv3 can hold all of this and much more all in one subnet)



Multi-tenant App Service recommendation



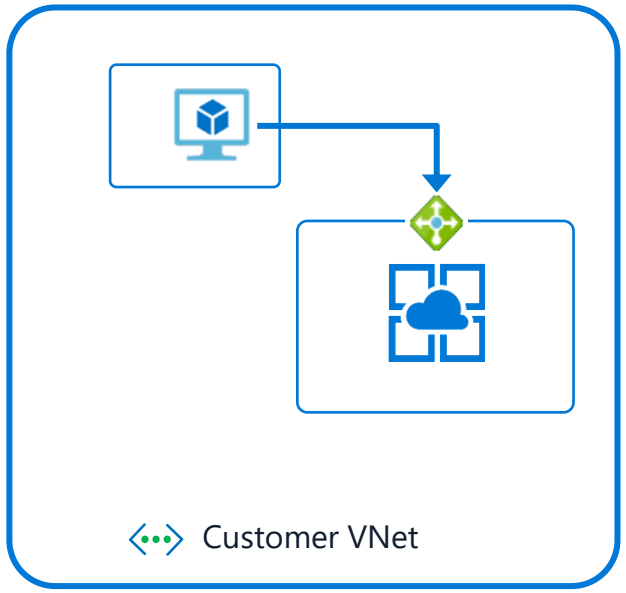
For an app in one region:

- Use private endpoints for inbound
- Use a WAF such as Application Gateway for inbound
- Enable VNet Integration for outbound
- Use NAT Gateway or Azure Firewall for outbound

For the same app in multiple regions:

- Use Azure Front Door
- Secure your app to Azure Front Door with Access Restrictions
- Enable VNet Integration for outbound
- Use NAT Gateway or Azure Firewall for outbound

App Service Environment



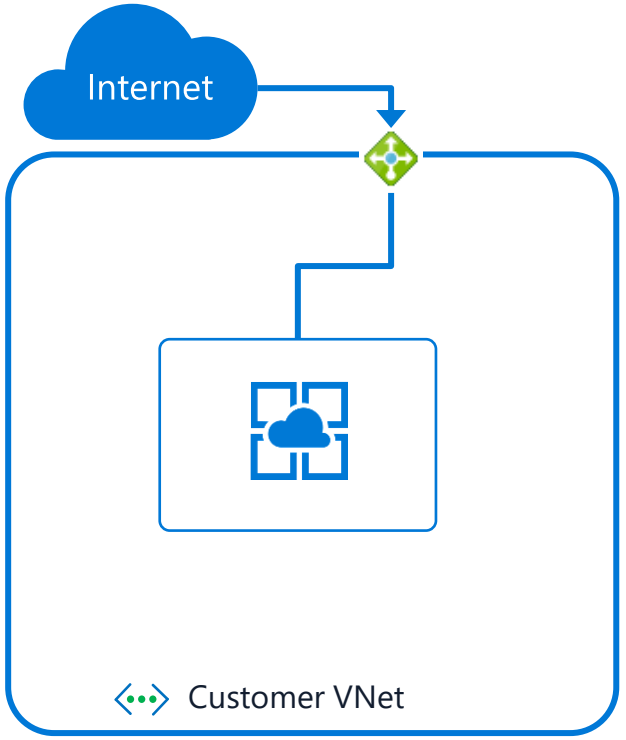
Internal VIP ASE

Single-tenant deployment of the App Service in your VNet

Can host your Windows apps, Windows containers, Linux apps and Linux containers

- App Service plans scale up to 100 instances
- An ASE can hold up to 200 total ASP instances across all plans added together

Two options affecting how apps are reached:
- ASE with an Internal VIP
- ASE with an External VIP (public address)



External VIP ASE

When to use ASE

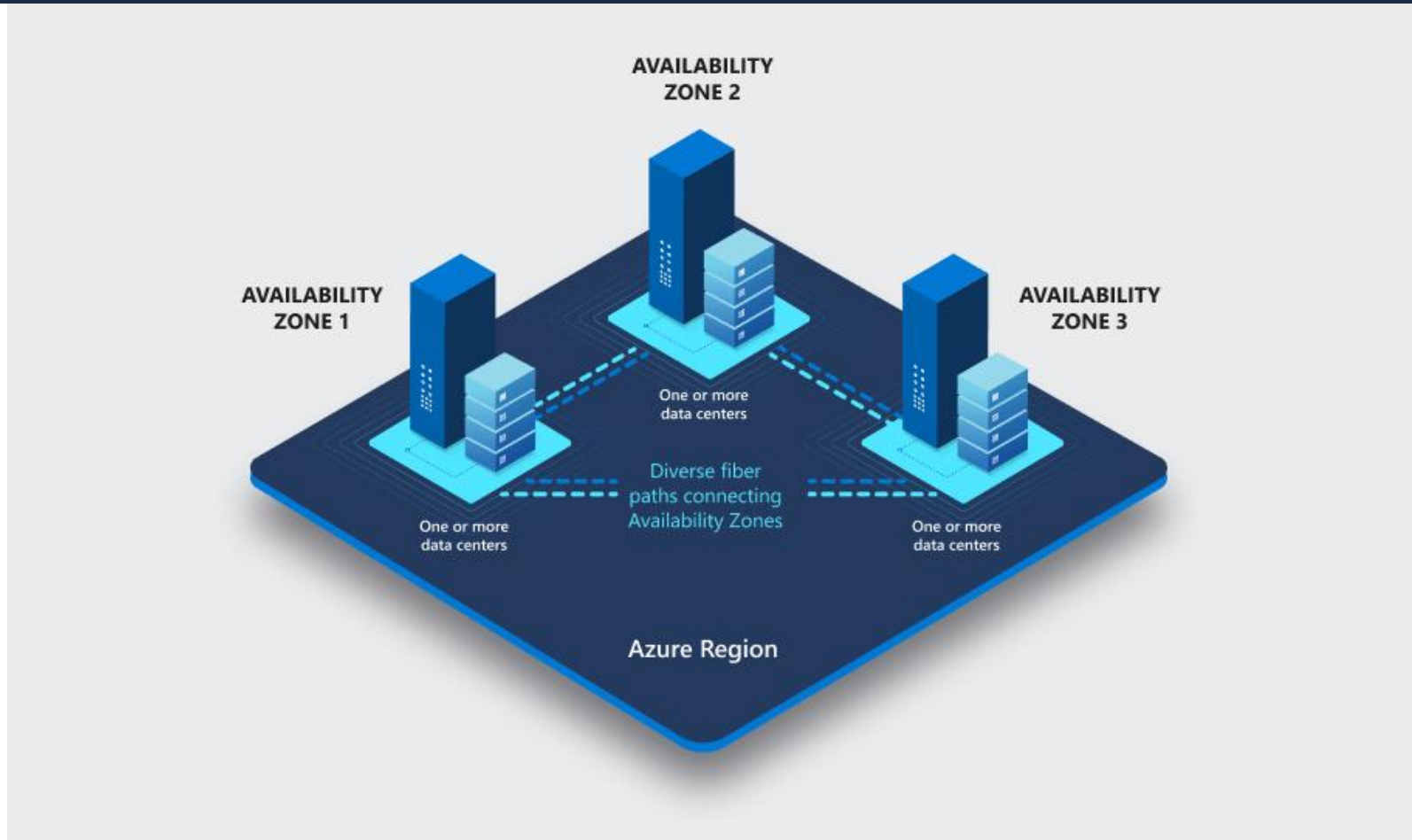
- High scale (more than 100 instances)
- Isolation
- High memory utilization
- High RPS
- Fast when it comes to VPN and on-premise connections

Examples

- Internal line-of-business applications
- Applications that need more than 30 App Service plan instances
- Single tenant system to satisfy internal compliance or security requirements
- Network isolated application hosting
- Multi-tier applications

Availability zones

- Physically separate locations within each Azure region
- Tolerant to local failures.
- Minimum of three separate availability zones are present in all availability zone-enabled regions.
- Connected by a high-performance network with a round-trip latency of less than 2ms.
- If one zone is affected, regional services, capacity, and high availability are supported by the remaining two zones.

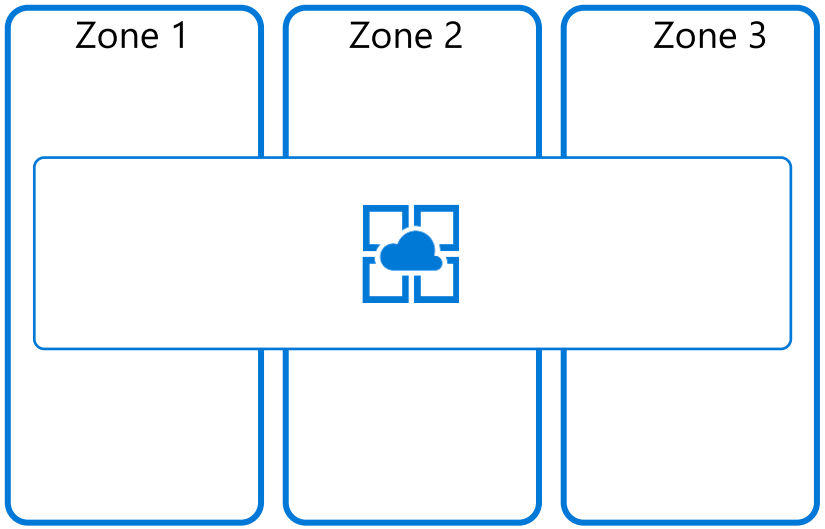


ASEv3 deployment types

Working to have normal and host group deployments in all regions
AZ deployments will only be in regions that support zone redundancy for our dependencies

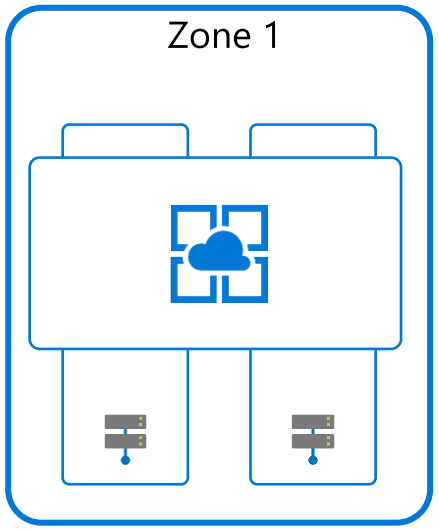
ASEv3

Normal deployment
1 zone



Availability Zone ASEv3

3 zones
Workloads are spread across all 3



Dedicated host ASEv3

1 zone
2 dedicated hosts

Zone Redundancy

- Only at Creation, Cannot changed
- All Service Plans will be zone redundant
- Supported only in specific regions
- Zone goes down, the App Service platform detects lost instances and automatically attempts to find new, replacement instances.
- Works with autoscaling
- Runtime behavior is always up
- Non-runtime behaviors might still be affected by an outage in other availability zones.
- No guarantee that requests for instances in a zone-down scenario will succeed

ASEv3 - pricing

Isolated v2 sizes: I1v2 (2 core 8 GB), I2v2 (4 core 16 GB), I3v2 (8 core 32 GB)

Isolated v2 Windows rate is ~$0.29/core hour.

Isolated v2 Linux rate is ~$0.21/core hour

(https://azure.microsoft.com/en-us/pricing/details/app-service/windows/)

Three deployment types, each with their own pricing model:

- **ASEv3**: If ASE is empty there is a charge as if you had one ASP with one instance of Windows I1v2. Not additive but only applied if totally empty, otherwise you just pay the Isolated v2 rate.
- **Availability Zone ASEv3**: Will require a minimum 9 Windows I1v2 instance charge. No added charge if you have 9 or more ASP instances. All ASPs have a minimum of three instances and are spread across AZ's as they scale out.
- **Dedicated host ASEv3**: Pay for two dedicated hosts per our pricing at ASEv3 creation then 20% Isolated V2 rate per core charge on top of that as you scale.

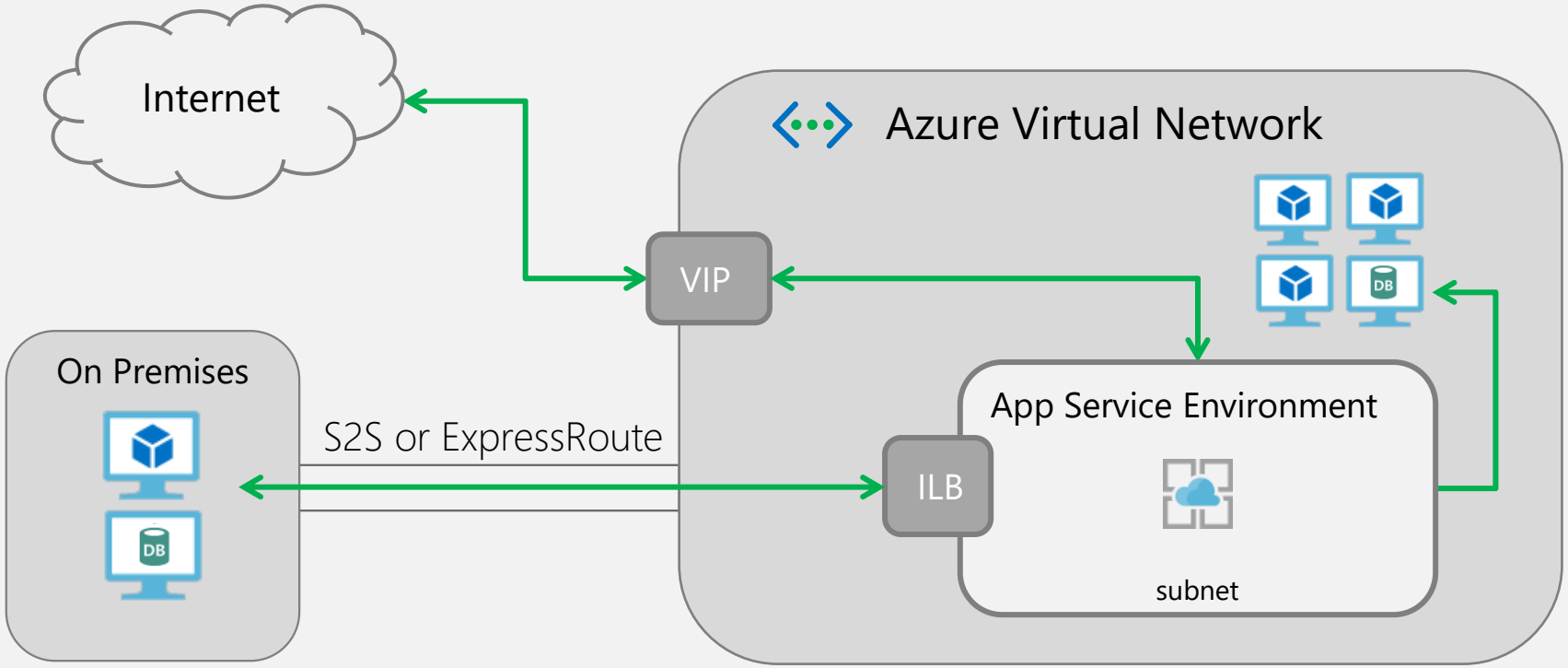
Reserved Instance pricing for Isolated v2 is available

ASE

- ✓ Apps can be exposed to internet through VIP
- ✓ ASE is deployed into a subnet within customer VNET
- ✓ VNET can be peered to other Azure resources or back on-prem

ASE Networking

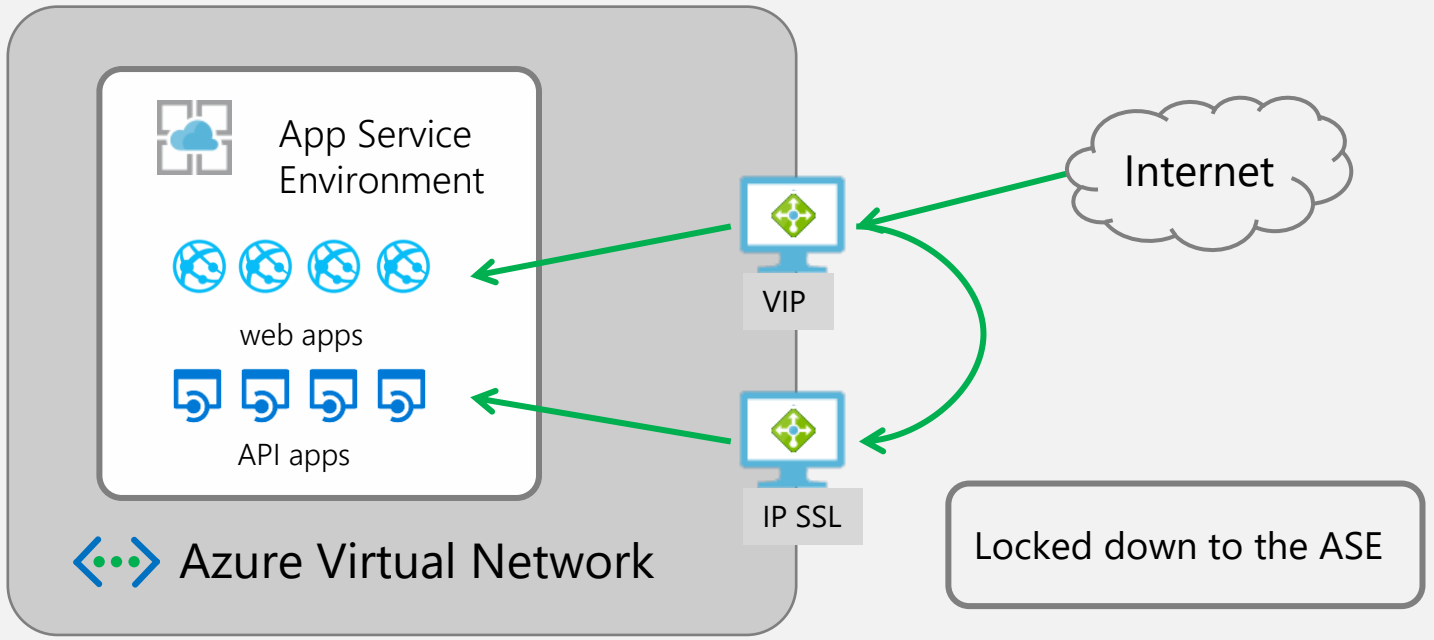
An ASE is a deployment of the Azure App Service into a subnet in a customer's Azure Virtual Network



ASE

- ✓ Public load balancer endpoint for accessing the web apps
- ✓ Use NSGs to lock down access to the app

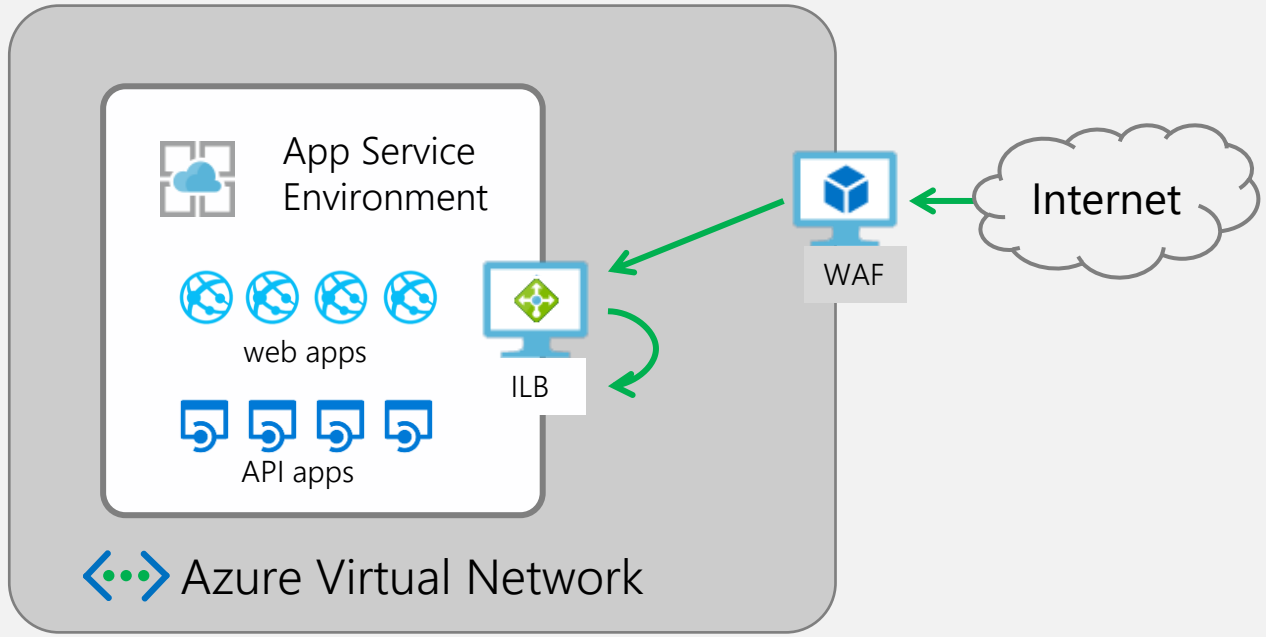
External ASE



ASE

- ✓ Protect public access with WAF functionality
- ✓ No public endpoints
- ✓ Traffic between web and APIs stays on VNET

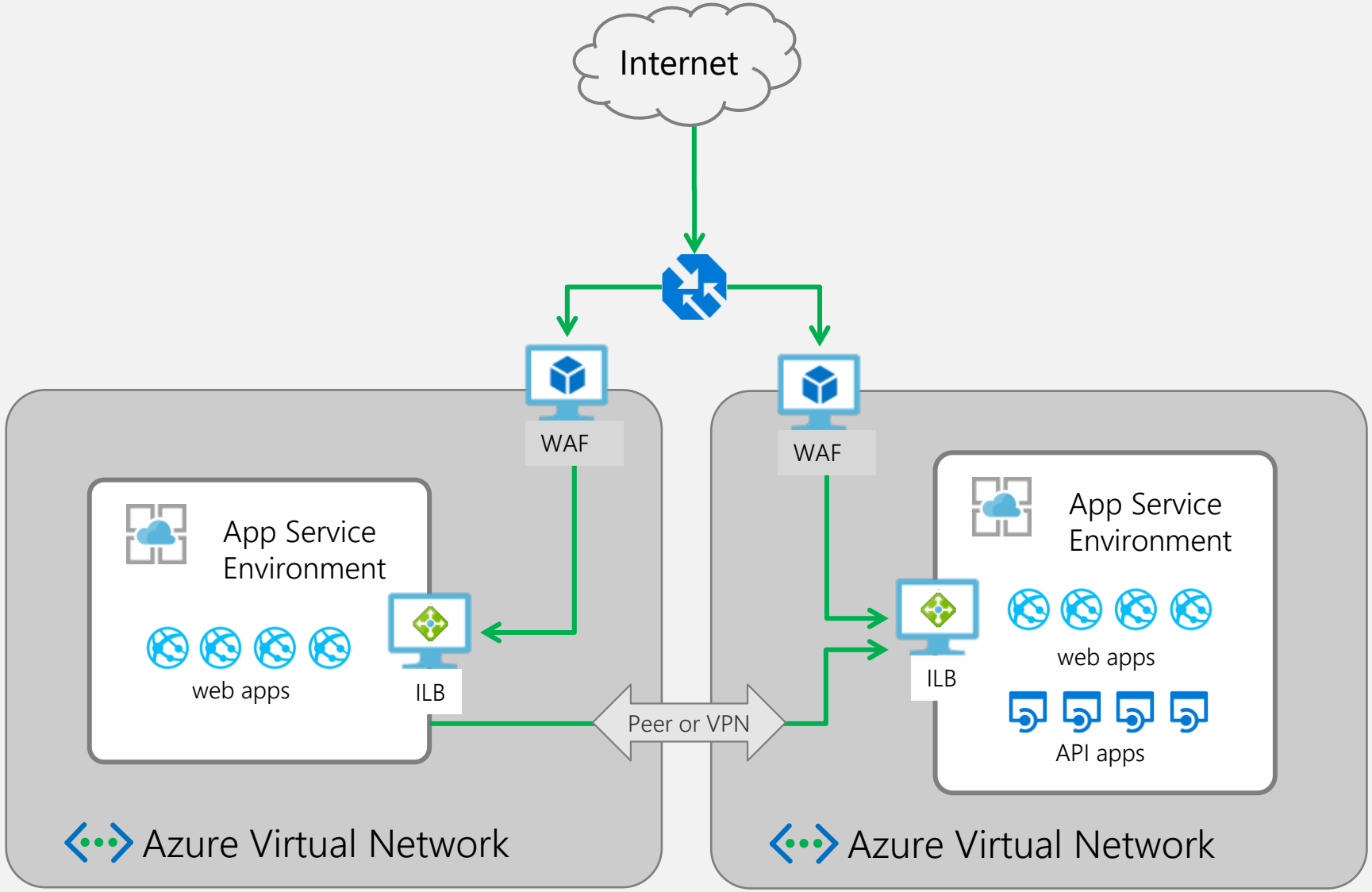
ILB ASE with WAF



ASE

- ✓ Protect public access with WAF functionality
- ✓ No public endpoints
- ✓ Traffic between web and APIs stays on VNET
- ✓ Geo distributed traffic patterns to minimize latency and maximize redundancy

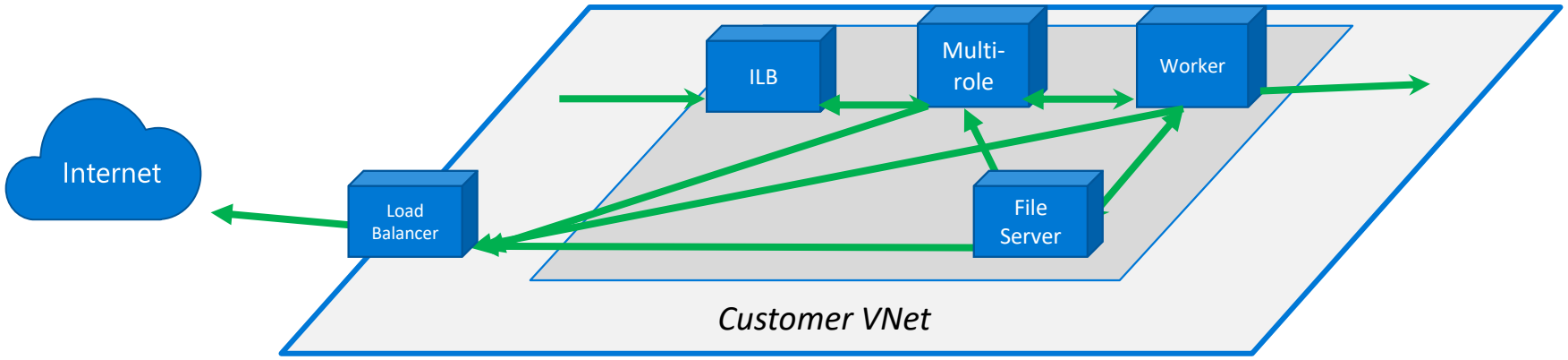
Geo Distributed ILB ASE



Internal or External VIP – when to use what

- **Internal VIP (ILB)**
  - For apps that should be WAF secured
  - For intranet applications that should only be accessible in/through the VNet
  - For backend API applications that should not be internet accessible
- **External VIP (ELB)**
  - Suits marketing sites and advertising campaigns
  - Exposing public services without a need for inbound security
  - Exposing geographically distributed apps behind an edge service like Azure Front Door

ASEv2 networking



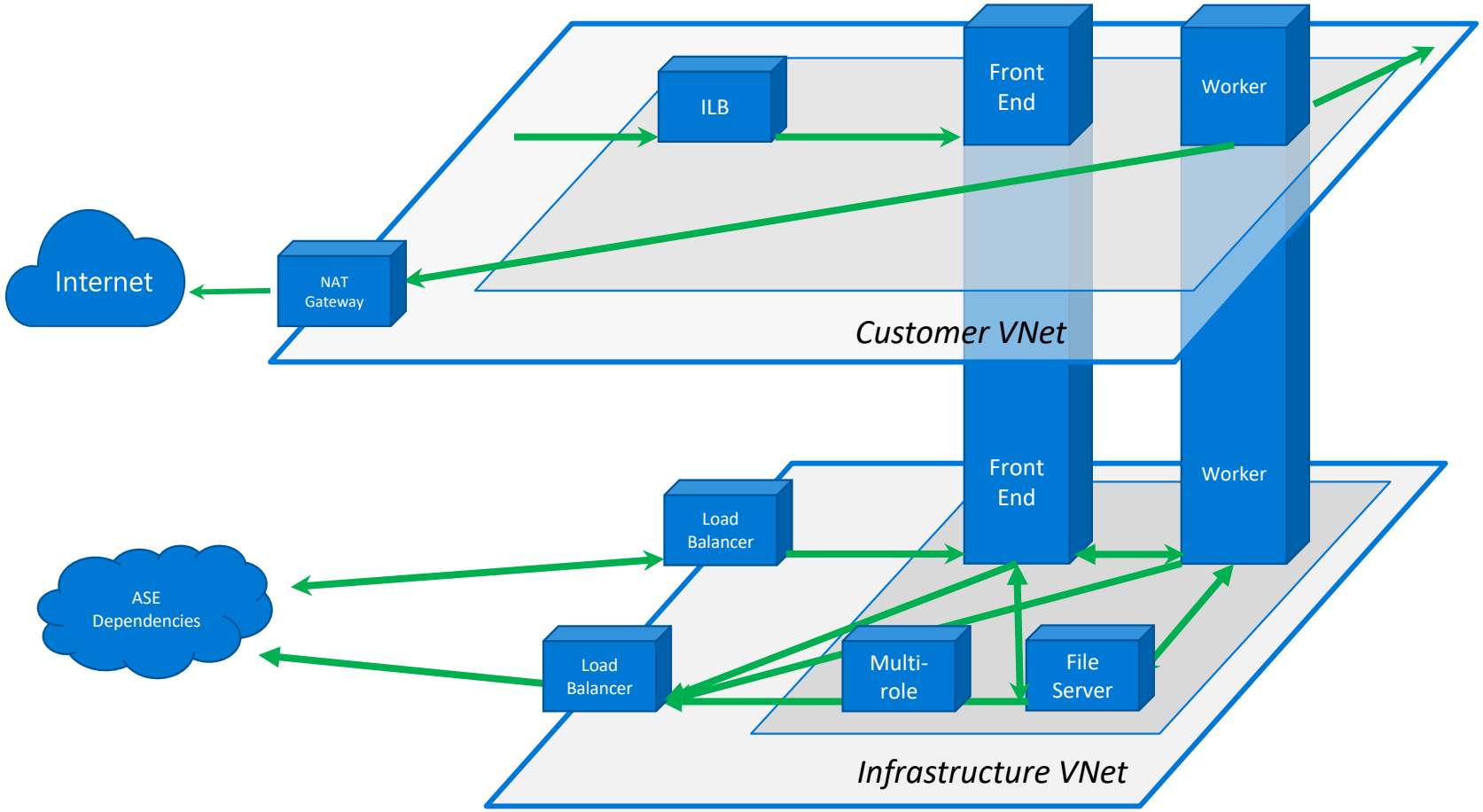
Customer traffic and ASE management traffic in the customer network.

Customer must allow for management traffic and ASE must allow customer traffic. Easy to break each other

Customers break ASEs with:

- NSGs
- Routes
- Firewall devices
- Interrupting SSL traffic

Multi-network approach - Internal VIP ASEv3



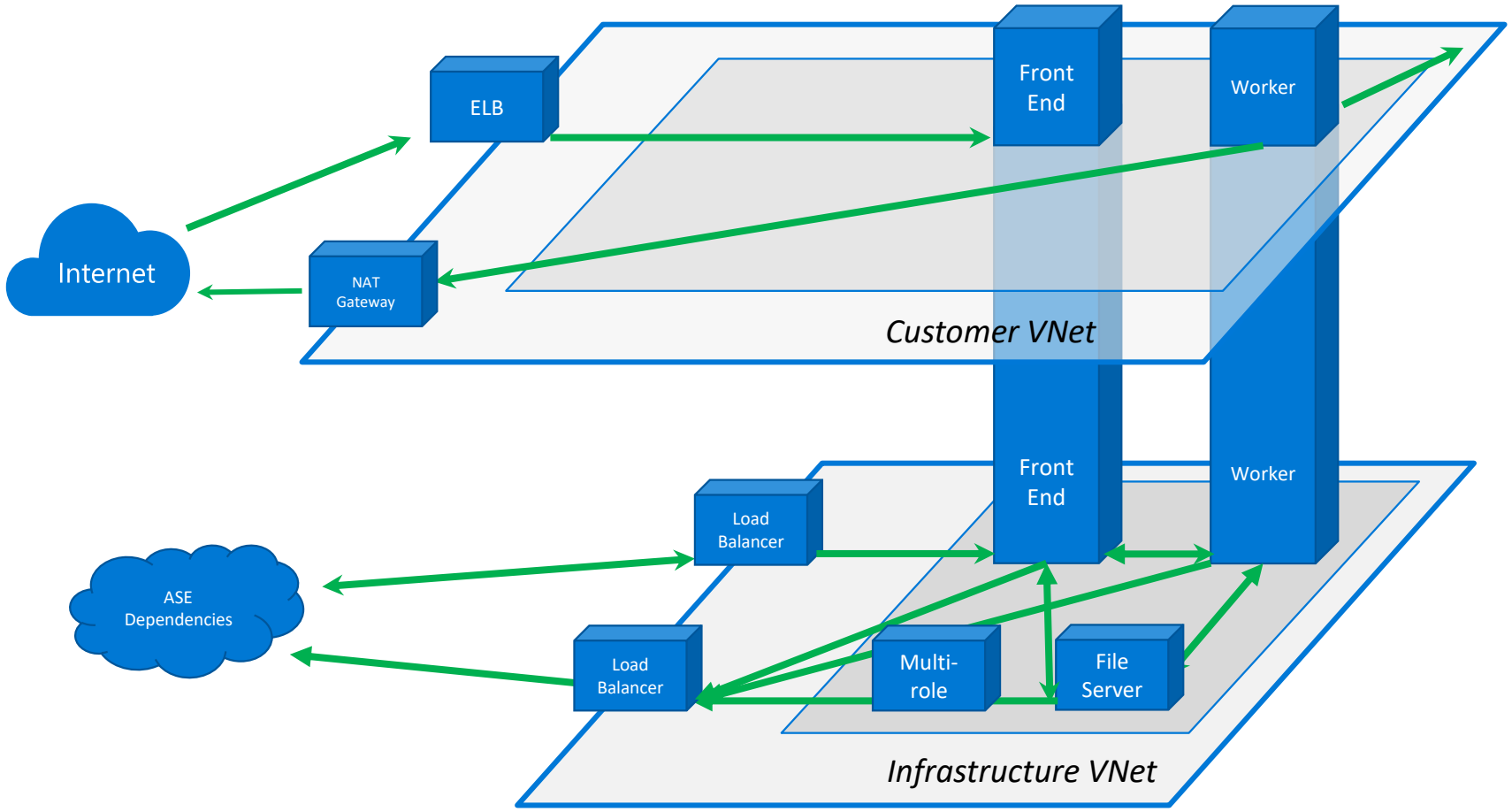
ASE is deployed into Microsoft managed VNet.

Inbound/outbound customer traffic is sent through separate interface with addresses in customer VNet/subnet.

Load balancer is provisioned in ASE subscription in customer VNet that can work with this injection technology.

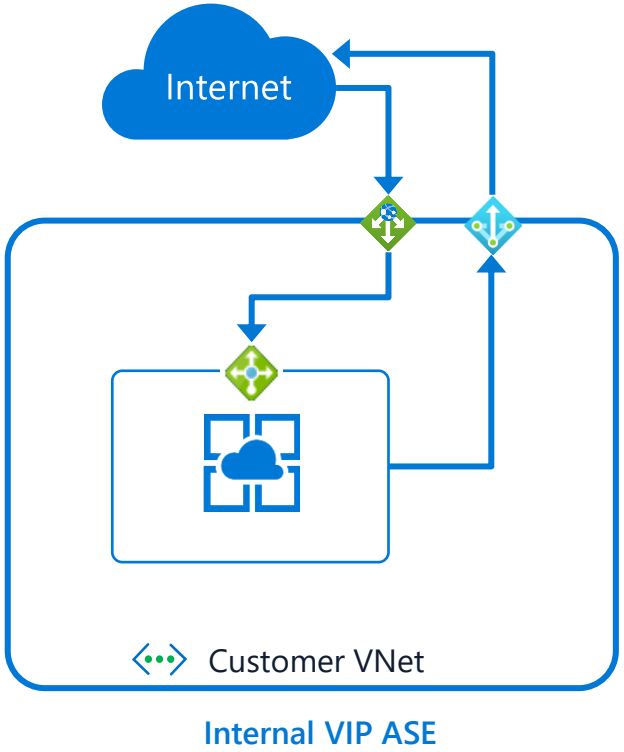
Control traffic in customer VNet as desired

Multi-network approach - External VIP ASEv3



Same as Internal VIP ASEv3 except the load balancer VIP is a public address

ASEv3 recommendation



To provide maximum security for internet accessible applications, use an Internal VIP ASEv3 with a WAF device, such as an Application Gateway, if you have internet exposed apps.

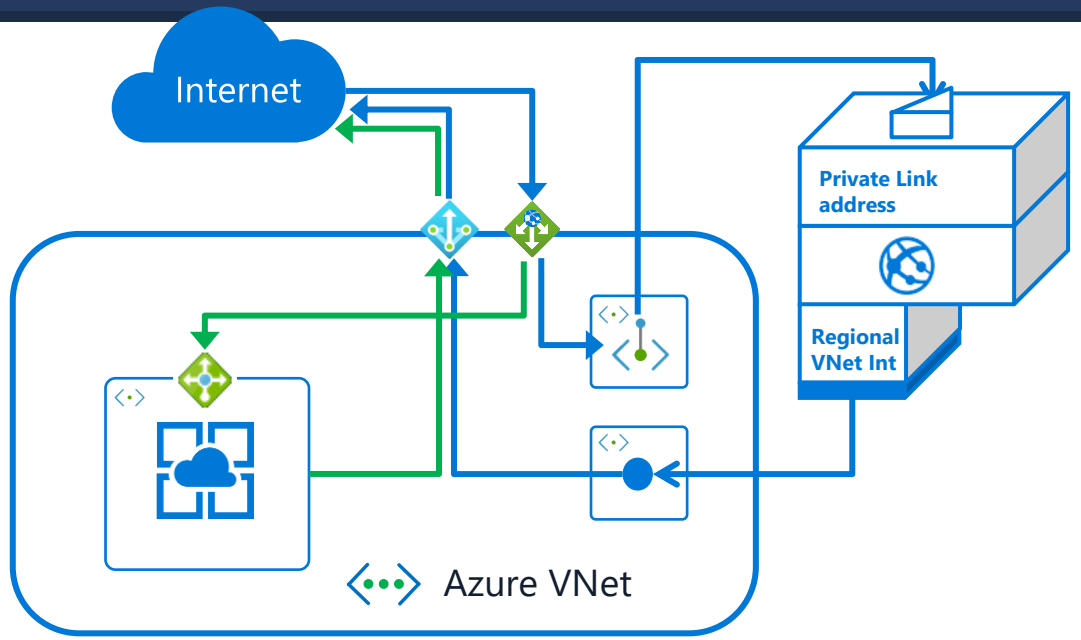
The default addresses with an ASEv3 exist for the life of the ASE. If you take it down and make it anew, you get new addresses. To ensure you can use your own address over and over or to handle any outbound SNAT problems, recommended to use a NAT Gateway.

ASEv3 compared to multi-tenant

Securing apps to a VNet in App Service

With ASEv3

- No features to enable on your apps
- Use NSGs to secure inbound and outbound
- Network isolation is controlled external to the app
- One subnet holds everything



With both

- Can route all outbound traffic as desired
- Can add WAF devices for inbound traffic
- Can add egress devices such as a NAT Gateway or Azure Firewall.

With multi-tenant

- Need to enable features on each app. Features can be disabled at the app level too.
- Can use NSGs to only secure outbound.
- Network isolation is controlled at each app
- Many subnets used for many apps

Operating differences

- **Multi-tenant**
  - Lower Pay-As-You-Go (PAYG) pricing
  - Scales out to 30 ASP instances
  - Fast scaling
  - Elastic premium support
  - Networking features enabled at app level
- **ASEv3**
  - Higher PAYG pricing but with Reserved Instance, cheaper than PremiumV2
  - Scales out to 100 ASP instances
  - Scaling isn't instant
  - No consumption plan options
  - No networking features needed on the apps

Securing apps in App Service

- **Multi-tenant**
  - Configure audit policy requiring apps to be private
  - Configure audit policy requiring apps to use VNet Integration
  - Add private endpoints on every app you want secured
  - Add VNet Integration to all apps you want secured
- **ASEv3**
  - Deploy apps to ASEv3
  - Pretend you are working
  - Look up latest cat videos online
  - Make dinner plans

Bottom line

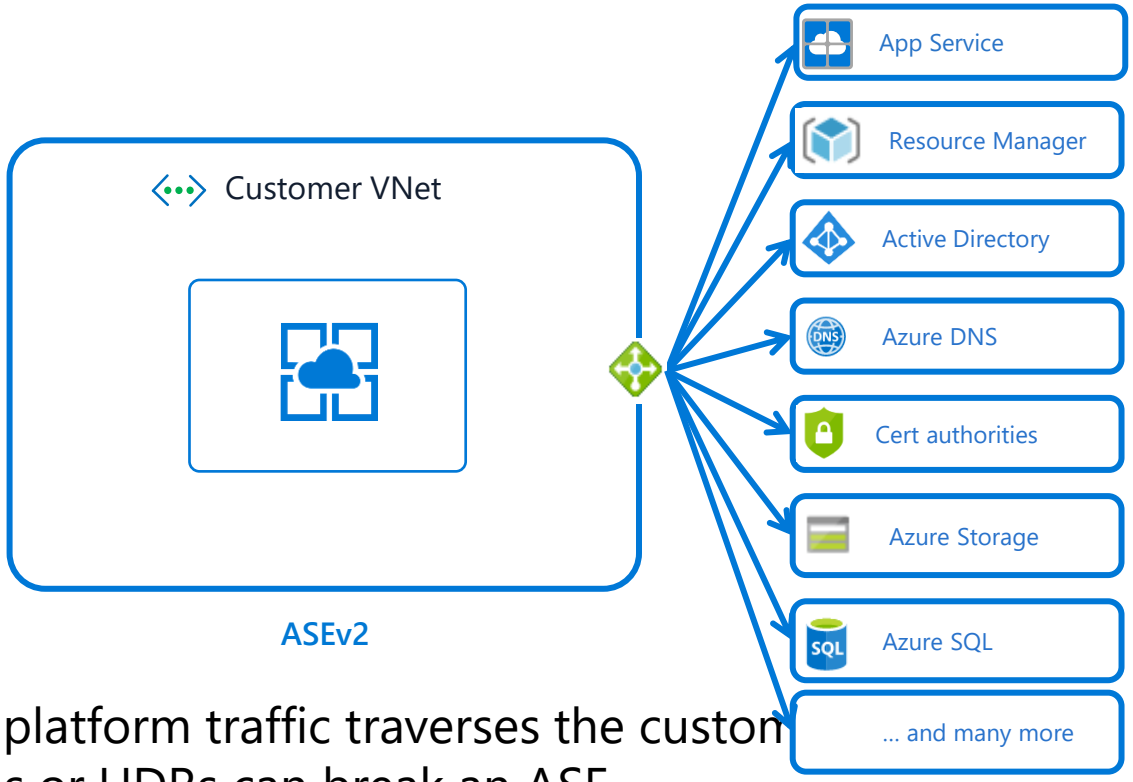
- ASEv3 provides the best security and isolation story in Azure. Better than VMs or any other service because there are no dependencies in the customer VNet.
- Instant scaling in the multi-tenant service is amazing. Go play with other service to compare if you like.

It all comes down to what is your top priority:

- Security and isolation – use ASEv3
- Fast scaling – use the multi-tenant App Service

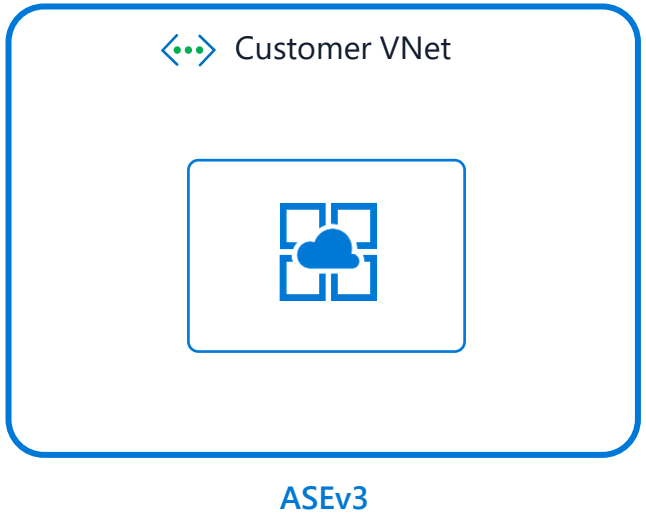
V3 vs V2

ASEv2 vs ASEv3 - network dependencies



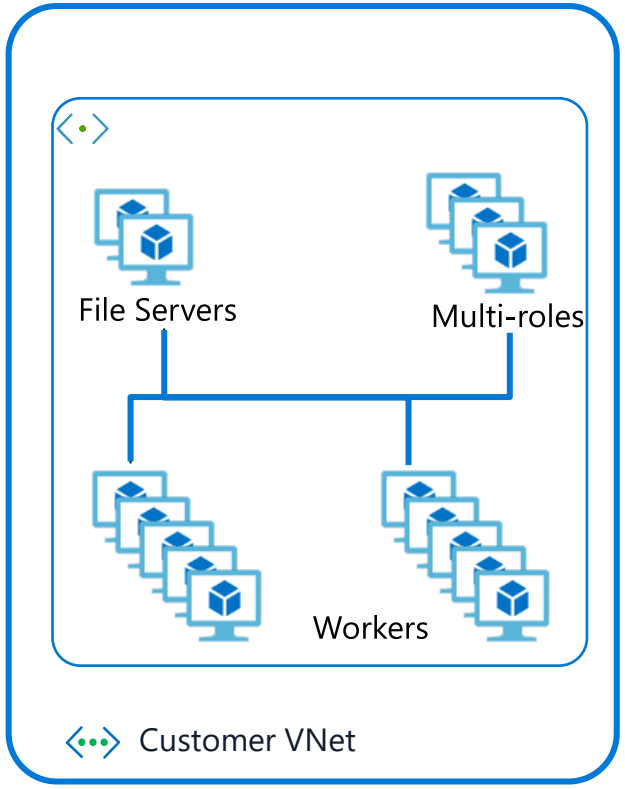
ASE platform traffic traverses the custom
NSGs or UDRs can break an ASE.

Customers can't block all internet traffic and must allow for dependency traffic.



No inbound or outbound management traffic travels on the customer VNet.

ASEv2 vs ASEv3 – compute layer



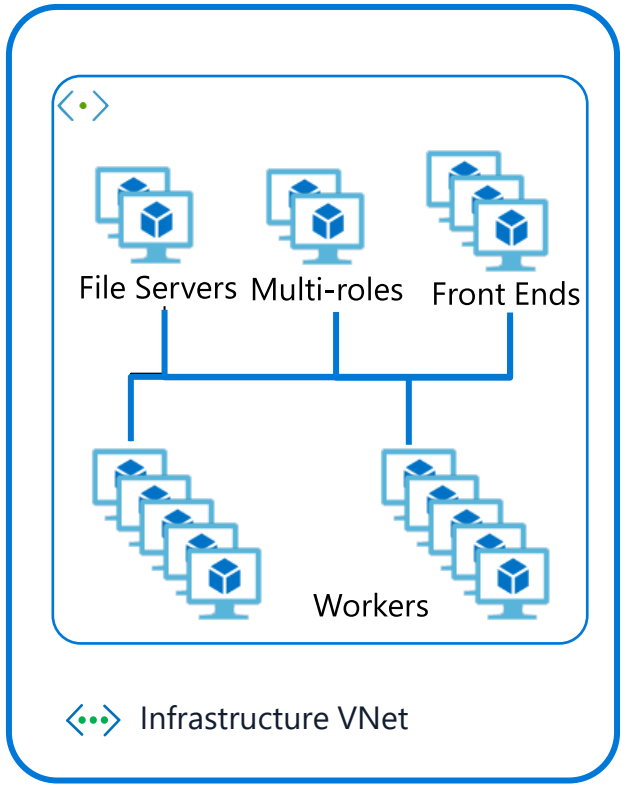
ASEv2

ASEv2:

- Cloud service based
- Dv2 VM based
- Front end is 1 core multi-role
- Scale operations block across the ASE

ASEv3:

- VMSS based
- Dv4 VM based
- Front end is 2 core dedicated front end
- Scale operations don't block other scale operations



ASEv3

ASEv2 vs ASEv3 - performance

- ASEv2
- ASE Deployment: 1 hour +
- Windows ASP scale: 30 min +
- Linux ASP scale: 1 hour +
- Throughput: X
- Sizes:
  1 core 3.5 GB
  2 core 7 GB
  4 core 14 GB

- ASEv3
- ASE Deployment: Currently ~2 hours, soon to be < 30 min
- Windows ASP scale: < 20 min*
- Linux ASP scale: < 15 min
- Throughput: 6X can burst up to 20X
- Sizes:
  2 core 8 GB
  4 core 16 GB
  8 core 32 GB

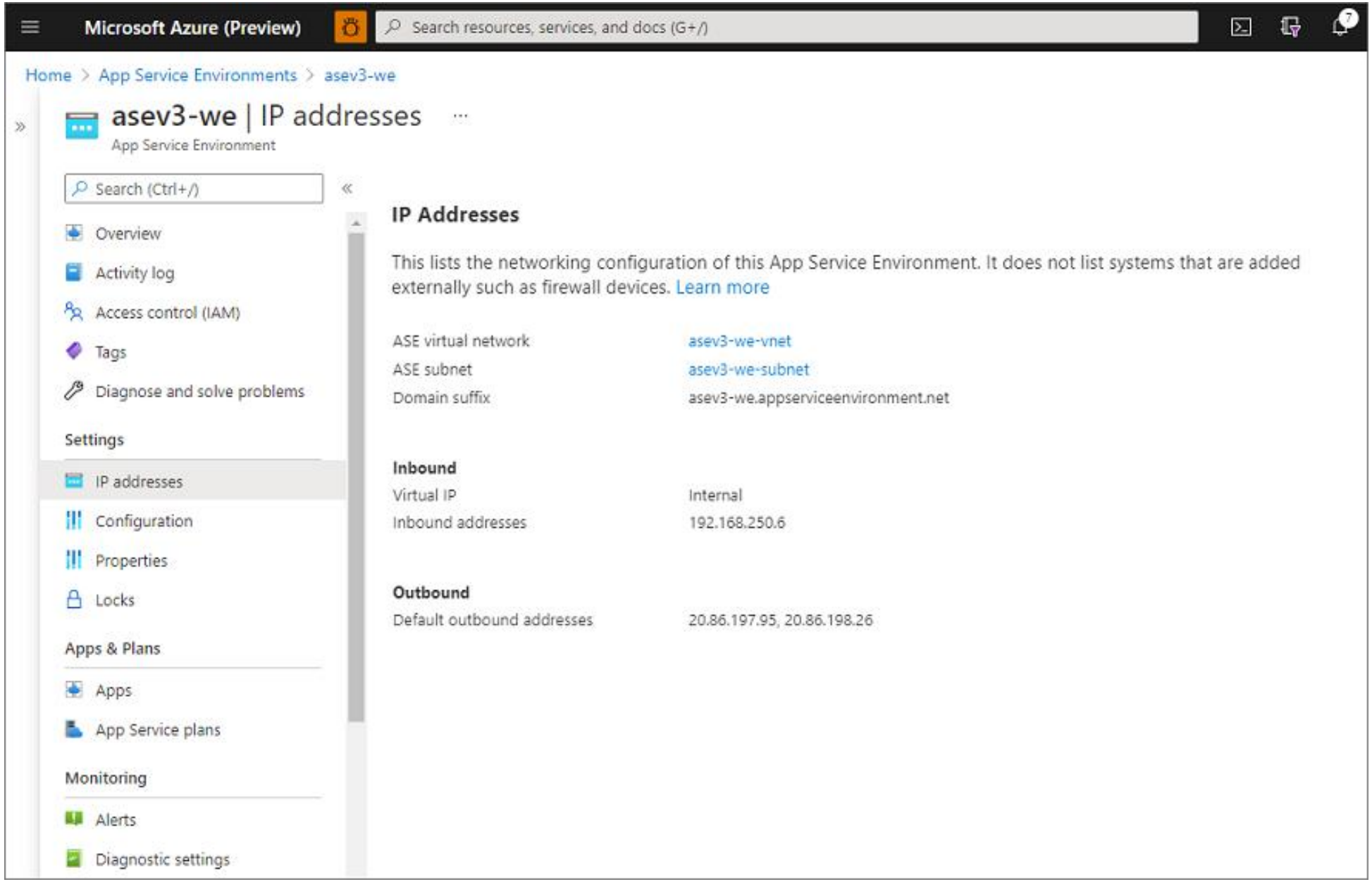
ASEv2 vs ASEv3 - other

- ASEv2
- Expensive: stamp fee and Isolated SKU rate
- Complex: many networking dependencies to track
- Scaling time indeterminate: Scaling out is 30 min+. Linux double that
- Rigid: hard to augment with new VM families

- ASEv3
- Inexpensive: no stamp fee and Isolated v2 SKU rate has RI
- Simple: no networking dependencies in the customer VNet.
- Scaling is reliably the same amount of time and Linux is <15 min
- Flexible: Due to the infrastructure VNet and VMSS, easier to make other changes like more VM families

Subnet requirements

- Dedicated empty sub net for **Microsoft.Web/hostingEnvironments**,
- Subnet size can affect scaling
- /24 address space (256 addresses) recommended
- Five addresses reserved for management purposes.
- ASE scale the dynamically scales the supporting infrastructure, and uses between 4 and 27 addresses
- The minimal size of your subnet is a /27 address space (32 addresses).
- You cannot scale out if you ran out of addresses
- Can experience increased latency during intensive traffic load, if Microsoft isn't able to scale the supporting infrastructure.


Microsoft Azure (Preview)
Search resources, services, and docs (G+/)
Home > App Service Environments > asev3-we
asev3-we | IP addresses
App Service Environment
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Settings
IP addresses
Configuration
Properties
Locks
Apps & Plans
Apps
App Service plans
Monitoring
Alerts
Diagnostic settings
IP Addresses
This lists the networking configuration of this App Service Environment. It does not list systems that are added externally such as firewall devices. Learn more
ASE virtual network
asev3-we-vnet
ASE subnet
asev3-we-subnet
Domain suffix
asev3-we.appserviceenvironment.net
Inbound
Virtual IP
Internal
Inbound addresses
192.168.250.6
Outbound
Default outbound addresses
20.86.197.95, 20.86.198.26

Network considerations

- Control inbound traffic from the NSG
- Load Balancer must have access to the ASE subnet or ASE will not be operational
- You can tunnel your outbound traffic through an egress firewall
- You can expose your apps through app gateway

- **External**
  - ASE is automatically on public DNS

  - **Internal**
  - You either have to use Azure Private DNS
  - Use your own DNS
  - You can configure some apps to use different DNS

Current ASEv3 GA limitations

There are a few limitations with ASEv3

- FTP
- Remote debug
- Sending traffic over port 25
- Monitoring traffic with NSG Flow, Network Watcher

All are in active development

Upgrade from ASEv2 to ASEv3 is in active development. Working to deliver it so you can upgrade from ASEv1 or ASEv2 up to ASEv3.

Module 2 Labs

Lab 2: Add Resources to the ASE

Exercise 1: Create an Application Gateway

Exercise 2: Create a Virtual Machine (Jumpbox)

Exercise 3: Create an App Service for a Web App

Questions?