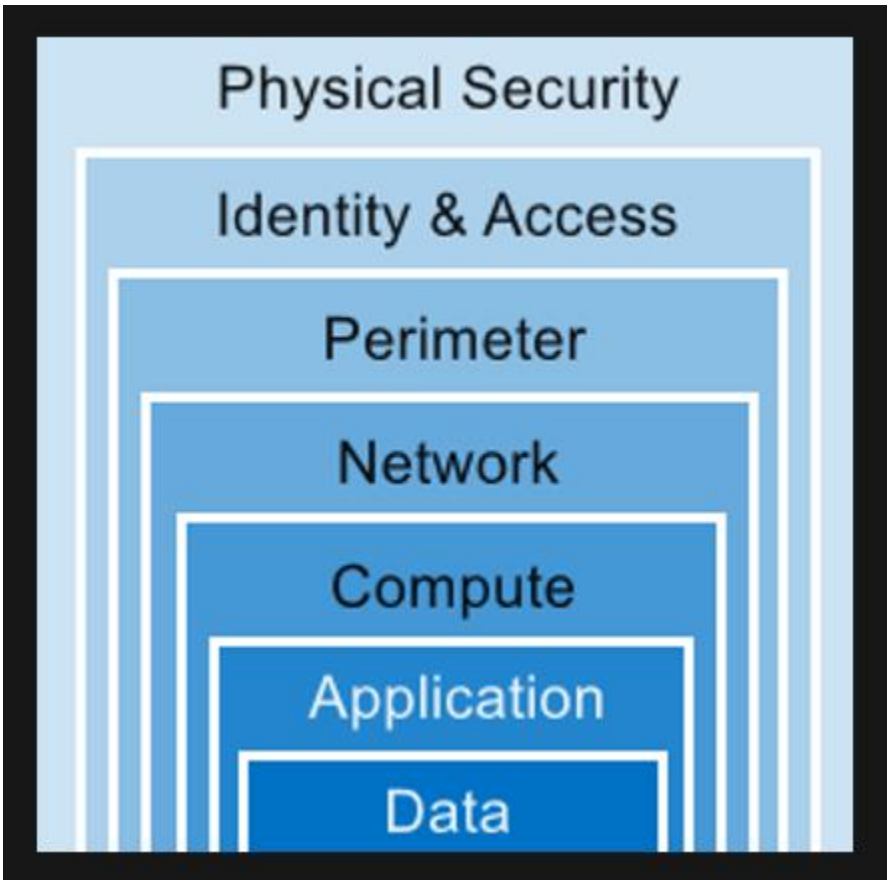
Defense in Depth



Network Micro-Segmentation

Micro-segmentation is a way to create secure zones in data centers and Azure deployments that allow you to isolate workloads and protect them individually. Security policies in a virtual environment can be assigned to virtual connections that can move with an application if the network is reconfigured – making the security policy persistent.

A best practice recommendation is to adopt a Zero Trust strategy based on user, device, and application identities. In contrast to network access controls that are based on elements such as source and destination IP address, protocols, and port numbers, Zero Trust enforces and validates access control at "access time". Granting access at access-time avoids the need to play a prediction game for an entire deployment, network, or subnet – only the destination resource needs to provide the necessary access controls.

- **Azure Network Security Groups** can be used for basic layer 3 & 4 access controls between Azure Virtual Networks, their subnets, and the Internet.
- **Application Security Groups** enable you to define fine-grained network security policies based on workloads, centralized on applications, instead of explicit IP addresses.

- **Azure Web Application Firewall** and the **Azure Firewall** can be used for more advanced network access controls that require application layer support.
- **Local Admin Password Solution (LAPS)** or a third-party Privileged Access Management can set strong local admin passwords and just in time access to them.

Additionally, third parties offer micro-segmentation approaches that may enhance your network controls by applying zero trust principles to networks you control with legacy assets on them.

Azure DDoS Protection

Azure Distributed Denial of Service (DDoS) protection, combined with application design best practices, provide defense against DDoS attacks. Azure DDoS protection provides the following service tiers:

- **Basic**: Automatically enabled as part of the Azure platform. Always-on traffic monitoring, and real-time mitigation of common network-level attacks, provide the same defenses utilized by Microsoft's online services. The entire scale of Azure's global network can be used to distribute and mitigate attack traffic across regions. Protection is provided for IPv4 and IPv6 Azure public IP addresses.
- **Standard**: Provides additional mitigation capabilities over the Basic service tier that are tuned specifically to Azure Virtual Network resources. DDoS Protection Standard is simple to enable, and requires no application changes. Protection policies are tuned through dedicated traffic monitoring and machine learning algorithms. Policies are applied to public IP addresses associated to resources deployed in virtual networks, such as Azure Load Balancer, Azure Application Gateway, and Azure Service Fabric instances, but this protection does not apply to App Service Environments. Real-time telemetry is available through Azure Monitor views during an attack, and for history. Rich attack mitigation analytics are available via diagnostic settings. Application layer protection can be added through the Azure Application Gateway Web Application Firewall or by installing a 3rd party firewall from Azure Marketplace. Protection is provided for IPv4 and IPv6 Azure public IP addresses.

Types of denial-of-service attacks that Azure protection mitigates

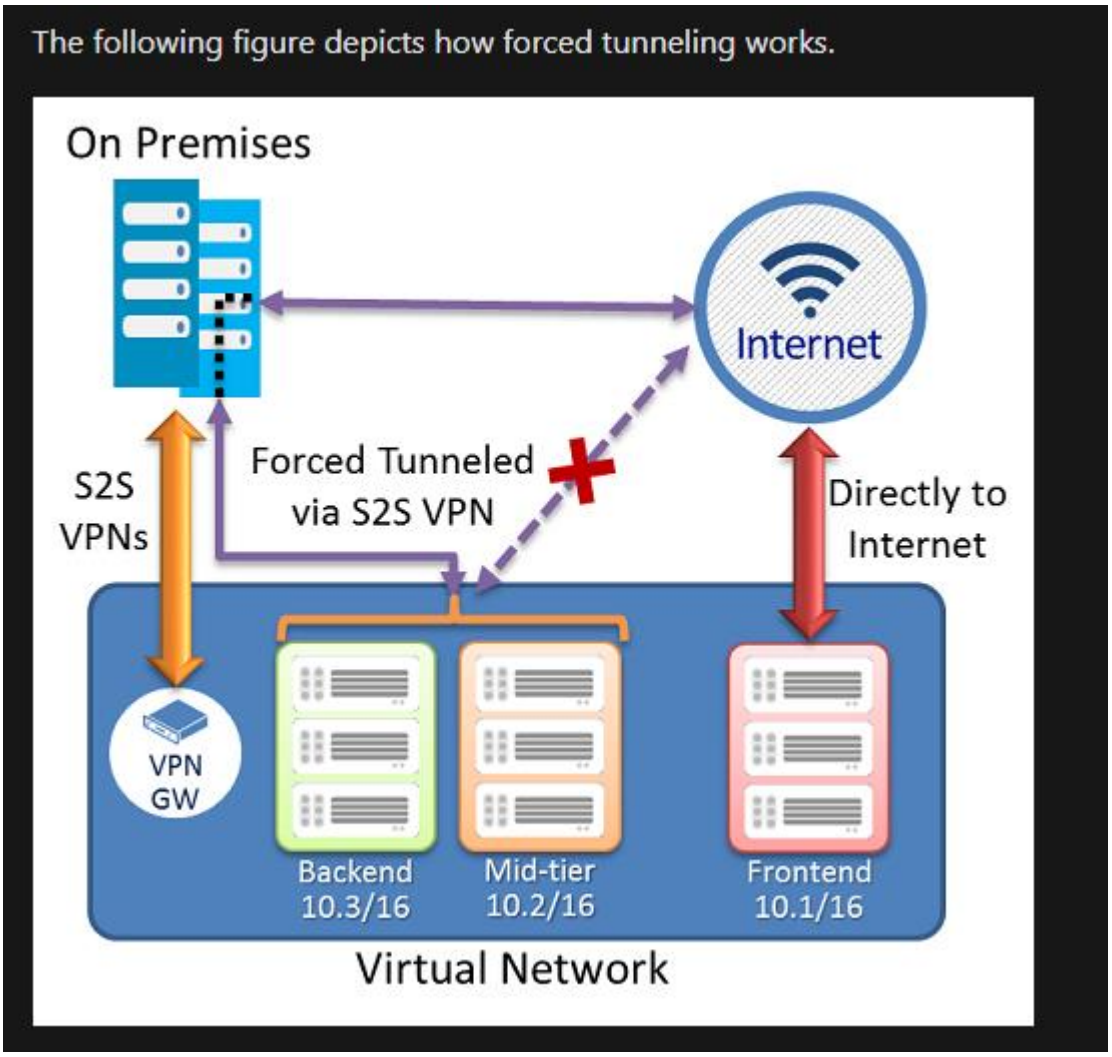
DDoS Protection Standard can mitigate the following types of attacks:

- **Volumetric attacks**: The attack's goal is to flood the network layer with a substantial amount of seemingly legitimate traffic. It includes UDP floods, amplification floods, and other spoofed-packet floods. DDoS Protection Standard mitigates these potential multi-gigabyte attacks by absorbing and scrubbing them, with Azure's global network scale, automatically.
- **Protocol attacks**: These attacks render a target inaccessible, by exploiting a weakness in the layer 3 and layer 4 protocol stack. It includes, SYN flood attacks, reflection attacks, and other protocol attacks. DDoS Protection Standard mitigates these attacks, differentiating between malicious and legitimate traffic, by interacting with the client, and blocking malicious traffic.
- **Resource (application) layer attacks**: These attacks target web application packets, to disrupt the transmission of data between hosts. The attacks include HTTP protocol violations, SQL

injection, cross-site scripting, and other layer 7 attacks. Use a Web Application Firewall, such as the Azure Application Gateway web application firewall, as well as DDoS Protection Standard to provide defense against these attacks. There are also third-party web application firewall offerings available in the Azure Marketplace.

Important

DDoS Protection Standard protects resources in a virtual network including public IP addresses associated with virtual machines, load balancers, and application gateways. When coupled with the Application Gateway web application firewall, or a third-party web application firewall deployed in a virtual network with a public IP, DDoS Protection Standard can provide full layer 3 to layer 7 mitigation capability.



In the preceding figure, the front-end subnet doesn't use forced tunneling. The workloads in the front-end subnet can continue to accept and respond to customer requests that come directly from the internet. The mid-tier and back-end subnets use forced tunneling. Any outbound connections from these two subnets to the internet are forced back to an on-premises site via one of the S2S VPN tunnels.

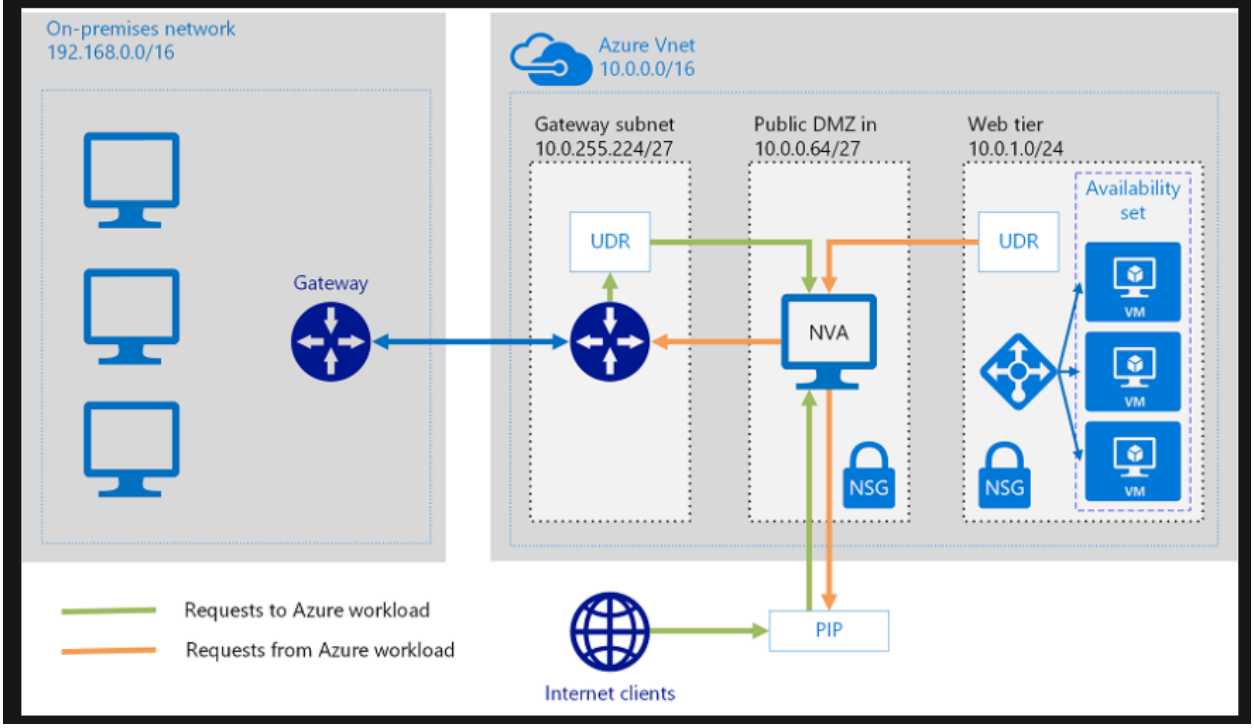
This allows you to restrict and inspect internet access from your VMs or cloud services in Azure while continuing to enable your multi-tier service architecture. If no internet-facing workloads exist in your VMs, you can also apply forced tunneling to the entire virtual network.

You configure forced tunneling in Azure via virtual network User Defined Routes (UDR). Redirecting traffic to an on-premises site is expressed as a default route to the Azure VPN gateway. This example uses UDRs to create a routing table to first add a default route and then associate the routing table with your virtual network subnets to enable forced tunneling on those subnets.

User Defined Routes

A User Defined Routes (UDR) is a custom route in Azure that overrides Azure's default system routes or adds routes to a subnet's route table. In Azure, you create a route table and then associate that route table with zero or more virtual network subnets. Each subnet can have zero or one route table associated with it. If you create a route table and associate it to a subnet, Azure either combines its routes with the default routes that Azure adds to a subnet or overrides those default routes.

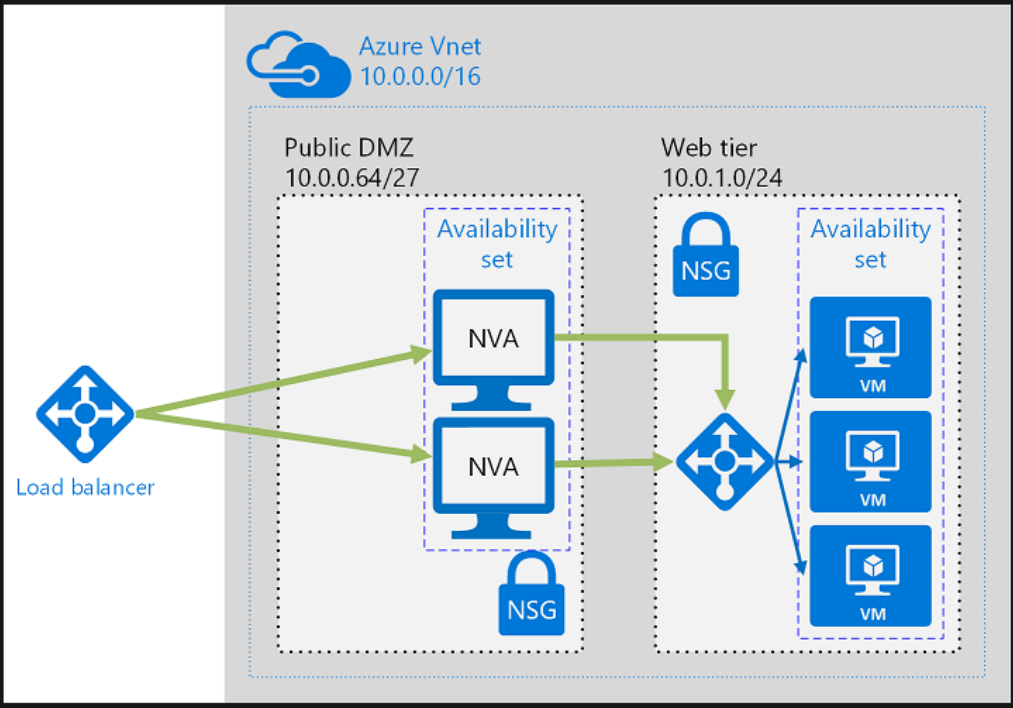
In this diagram UDRs are used to direct traffic from the Gateway subnet and the Web tier to the Network Virtual Appliance (NVA).



You can deploy an NVA to a perimeter network in many architectures. In the previous diagram, the NVA helps provide a secure network boundary by checking all inbound and outbound network traffic and then passing only the traffic that meets the network security rules. However, the fact that all network traffic passes through the NVA means that the NVA is a single point of failure in the network. If the NVA fails, no other path will exist for network traffic, and all the back-end subnets will become unavailable.

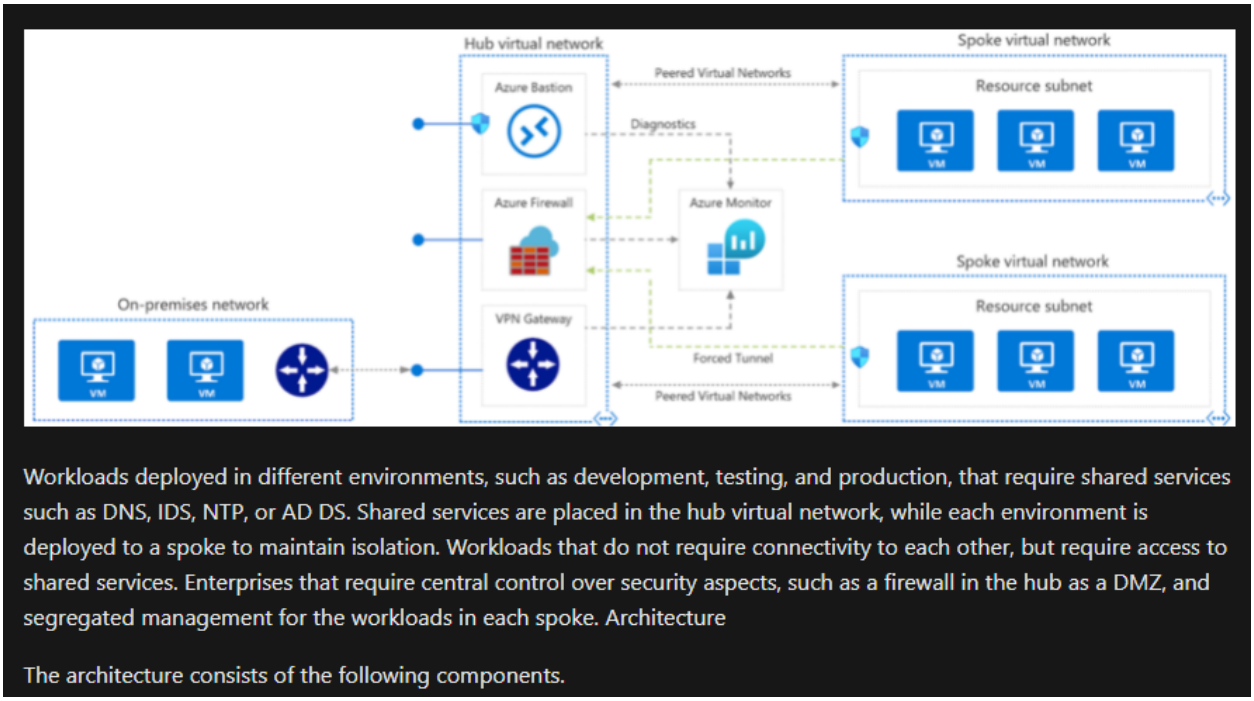
To make an NVA highly available, deploy more than one NVA into an availability set.

The following figure shows a high-availability architecture that implements an ingress perimeter network behind an internet-facing load balancer. This architecture is designed to provide connectivity to Azure workloads for layer 7 traffic, such as HTTP or HTTPS traffic.



The benefit of this architecture is that all NVAs are active, and if one fails, the load balancer directs network traffic to the other NVA. Both NVAs route traffic to the internal load balancer, so if one NVA is active, traffic will continue to flow. The NVAs are required to terminate SSL traffic intended for the web tier VMs. These NVAs can't be extended to handle on-premises traffic, because on-premises traffic requires another dedicated set of NVAs with their own network.

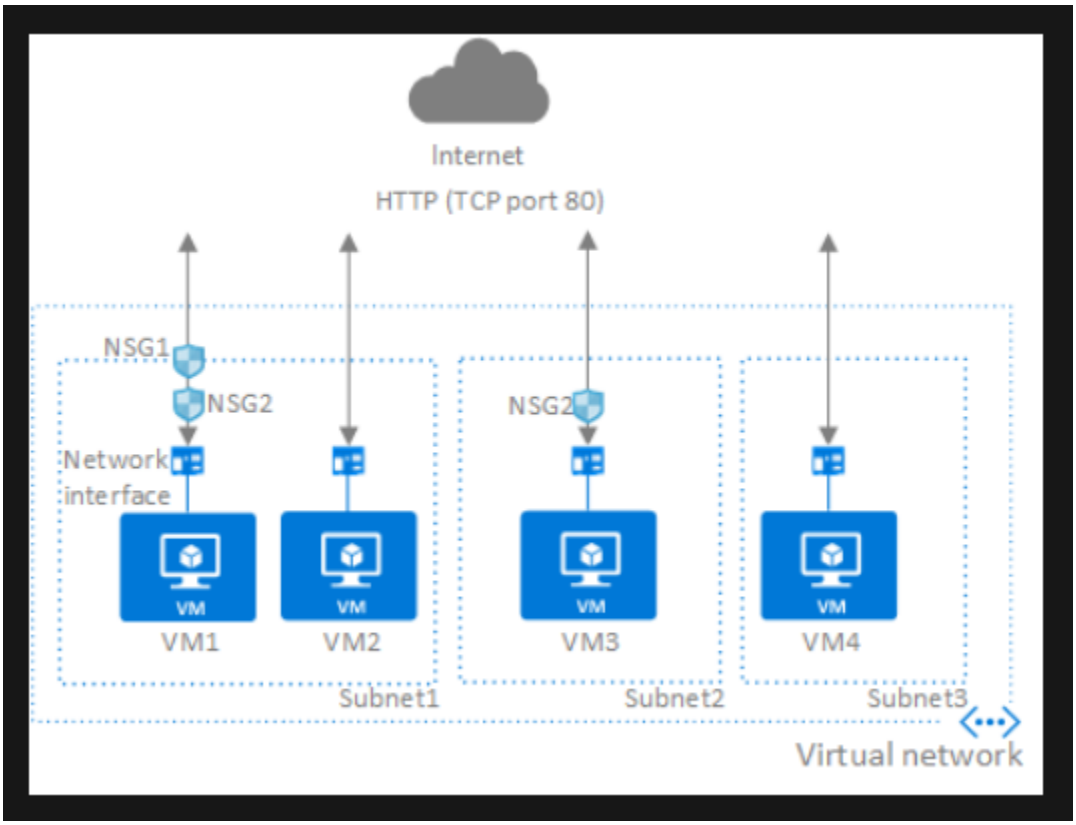
UDRs and NSGs help provide layer 3 and layer 4 (of the OSI model) security. NVAs help provide layer 7, application layer,



The architecture consists of the following components.

- **On-premises network** - A private local-area network running within an organization.
- **VPN device** - A device or service that provides external connectivity to the on-premises network. The VPN device may be a hardware device, or a software solution such as the Routing and Remote Access Service (RRAS) in Windows Server 2012. For a list of supported VPN appliances and information on configuring selected VPN appliances for connecting to Azure, see About VPN devices for Site-to-Site VPN Gateway connections.
- **VPN virtual network gateway or ExpressRoute gateway** - The virtual network gateway enables the virtual network to connect to the VPN device, or ExpressRoute circuit, used for connectivity with your on-premises network. For more information, see Connect an on-premises network to a Microsoft Azure virtual network.
- **Hub virtual network** - The virtual network used as the hub in the hub-spoke topology. The hub is the central point of connectivity to your on-premises network, and a place to host services that can be consumed by the different workloads hosted in the spoke virtual networks.
- **Gateway subnet** - The virtual network gateways are held in the same subnet.
- **Spoke virtual networks** - One or more virtual networks that are used as spokes in the hub-spoke topology. Spokes can be used to isolate workloads in their own virtual networks, managed separately from other spokes. Each workload might include multiple tiers, with multiple subnets connected through Azure load balancers. For more information about the application infrastructure, see Running Windows VM workloads and Running Linux VM workloads.
- **Virtual network peering** - Two virtual networks can be connected using a peering connection. Peering connections are non-transitive, low latency connections between virtual networks. Once

peered, the virtual networks exchange traffic by using the Azure backbone, without the need for a router. In a hub-spoke network topology, you use virtual network peering to connect the hub to each spoke. You can peer virtual networks in the same region, or different regions. For more information, see Requirements and constraints.



Reference the above diagram, along with the following text, to understand how Azure processes inbound and outbound rules for **network security groups**:

Inbound traffic

For inbound traffic, Azure processes the rules in a network security group associated to a subnet first, if there is one, and then the rules in a network security group associated to the network interface, if there is one.

- **VM1**: The security rules in NSG1 are processed, since it is associated to Subnet1 and VM1 is in Subnet1. Unless you've created a rule that allows port 80 inbound, the traffic is denied by the **DenyAllInbound default security rule**, and never evaluated by NSG2, since NSG2 is associated to the network interface. If NSG1 has a security rule that allows port 80, the traffic is then processed by NSG2. To allow port 80 to the virtual machine, both NSG1 and NSG2 must have a rule that allows port 80 from the internet.
- **VM2**: The rules in NSG1 are processed because VM2 is also in Subnet1. Since VM2 does not have a network security group associated to its network interface, it receives all traffic allowed

through NSG1 or is denied all traffic denied by NSG1. Traffic is either allowed or denied to all resources in the same subnet when a network security group is associated to a subnet.

- **VM3**: Since there is no network security group associated to Subnet2, traffic is allowed into the subnet and processed by NSG2, because NSG2 is associated to the network interface attached to VM3.
- **VM4**: Traffic is allowed to VM4, because a network security group isn't associated to Subnet3, or the network interface in the virtual machine. All network traffic is allowed through a subnet and network interface if they don't have a network security group associated to them.

Outbound traffic

For outbound traffic, Azure processes the rules in a network security group associated to a network interface first, if there is one, and then the rules in a network security group associated to the subnet, if there is one.

- **VM1**: The security rules in NSG2 are processed. Unless you create a security rule that denies port 80 outbound to the internet, the traffic is allowed by the AllowInternetOutbound default security rule in both NSG1 and NSG2. If NSG2 has a security rule that denies port 80, the traffic is denied, and never evaluated by NSG1. To deny port 80 from the virtual machine, either, or both of the network security groups must have a rule that denies port 80 to the internet.
- **VM2**: All traffic is sent through the network interface to the subnet, since the network interface attached to VM2 does not have a network security group associated to it. The rules in NSG1 are processed.
- **VM3**: If NSG2 has a security rule that denies port 80, the traffic is denied. If NSG2 has a security rule that allows port 80, then port 80 is allowed outbound to the internet, since a network security group is not associated to Subnet2.
- **VM4**: All network traffic is allowed from VM4, because a network security group isn't associated to the network interface attached to the virtual machine, or to Subnet3.

Intra-subnet traffic

It's important to note that security rules in an NSG associated to a subnet can affect connectivity between VM's within it. For example, if a rule is added to NSG1 which denies all inbound and outbound traffic, VM1 and VM2 will no longer be able to communicate with each other. Another rule would have to be added specifically to allow this.

General guidelines

Unless you have a specific reason to, we recommended that you associate a network security group to a subnet, or a network interface, but not both. Since rules in a network security group associated to a subnet can conflict with rules in a network security group associated to a network interface, you can have unexpected communication problems that require troubleshooting.

ASG

The rules that specify an ASG as the source or destination are only applied to the network interfaces that are members of the ASG. If the network interface is not a member of an ASG, the rule is not applied to the network interface even though the network security group is associated to the subnet.

Application security groups have the following constraints

- There are limits to the number of ASGs you can have in a subscription, in addition to other limits related to ASGs.
- You can specify one ASG as the source and destination in a security rule. You cannot specify multiple ASGs in the source or destination.
- All network interfaces assigned to an ASG must exist in the same virtual network that the first network interface assigned to the ASG is in. For example, if the first network interface assigned to an ASG named AsgWeb is in the virtual network named VNet1, then all subsequent network interfaces assigned to ASGWeb must exist in VNet1. You cannot add network interfaces from different virtual networks to the same ASG.
- If you specify an ASG as the source and destination in a security rule, the network interfaces in both ASGs must exist in the same virtual network. For example, if AsgLogic contained network interfaces from VNet1, and AsgDb contained network interfaces from VNet2, you could not assign AsgLogic as the source and AsgDb as the destination in a rule. All network interfaces for both the source and destination ASGs need to exist in the same virtual network.

Summary

Application Security Groups along with NSGs, have brought multiple benefits on the network security area:

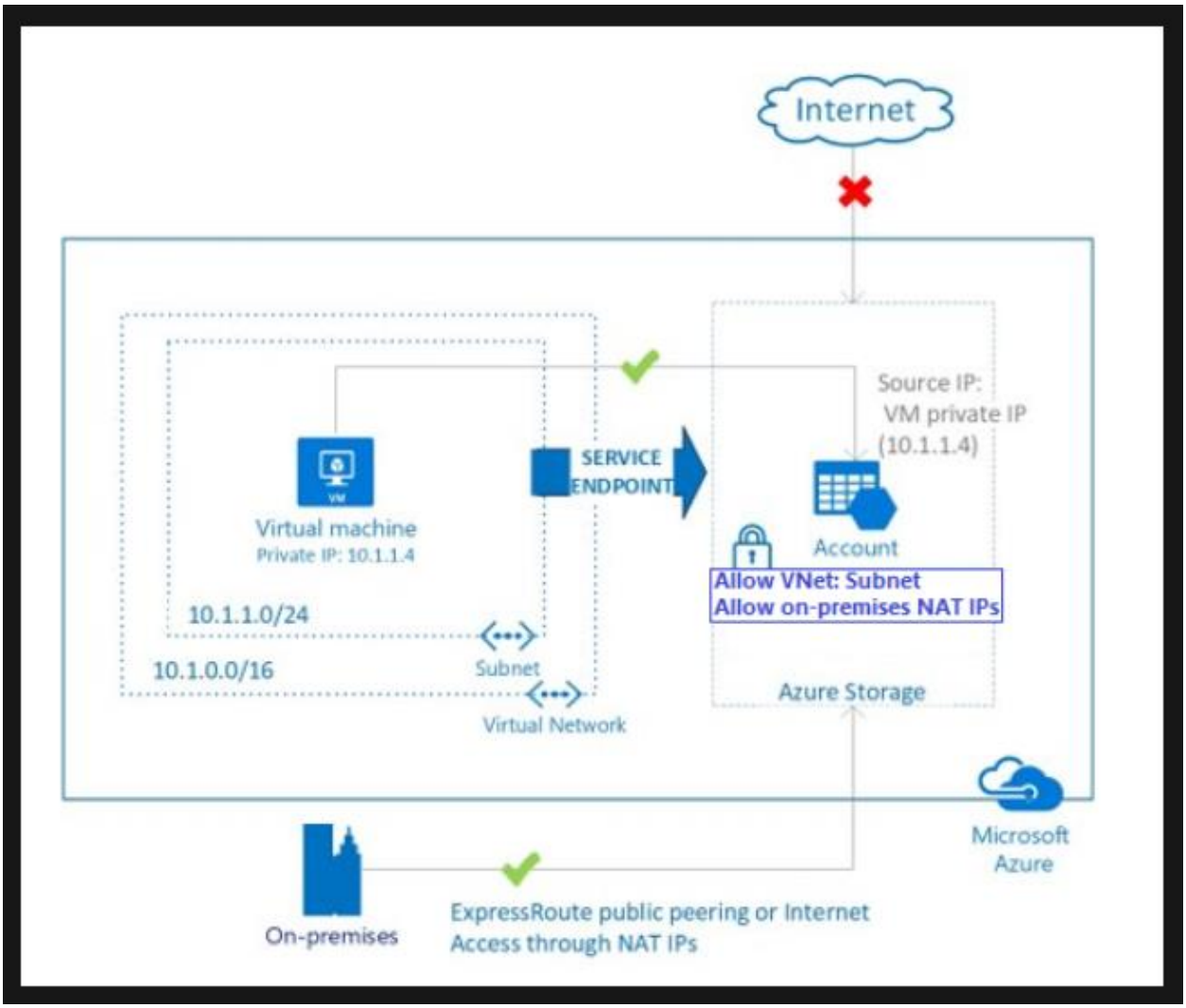
- **A single management experience**
- **Increased limits on multiple dimensions**
- **A great level of simplification**
- **A seamless integration with your architecture**

Enable service endpoints

A virtual network service endpoint provides the identity of your virtual network to the Azure service. Once service endpoints are enabled in your virtual network, you can secure Azure service resources to your virtual network by adding a virtual network rule to the resources.

Today, Azure service traffic from a virtual network uses public IP addresses as source IP addresses. With service endpoints, service traffic switches to use virtual network private addresses as the source IP addresses when accessing the Azure service from a virtual network. This switch allows you to access the services without the need for reserved, public IP addresses used in IP firewalls.

A common usage case for service endpoints is a virtual machine accessing storage. The storage account restricts access to the virtual machines private IP address.



Why use a service endpoint?

- **Improved security for your Azure service resources**. VNet private address space can be overlapping and so, cannot be used to uniquely identify traffic originating from your VNet. Service endpoints provide the ability to secure Azure service resources to your virtual network, by extending VNet identity to the service. Once service endpoints are enabled in your virtual network, you can secure Azure service resources to your virtual network by adding a virtual network rule to the resources. This provides improved security by fully removing public Internet access to resources, and allowing traffic only from your virtual network.
- **Optimal routing for Azure service traffic from your virtual network**. Today, any routes in your virtual network that force Internet traffic to your premises and/or virtual appliances, known as forced-tunneling, also force Azure service traffic to take the same route as the Internet traffic. Service endpoints provide optimal routing for Azure traffic.

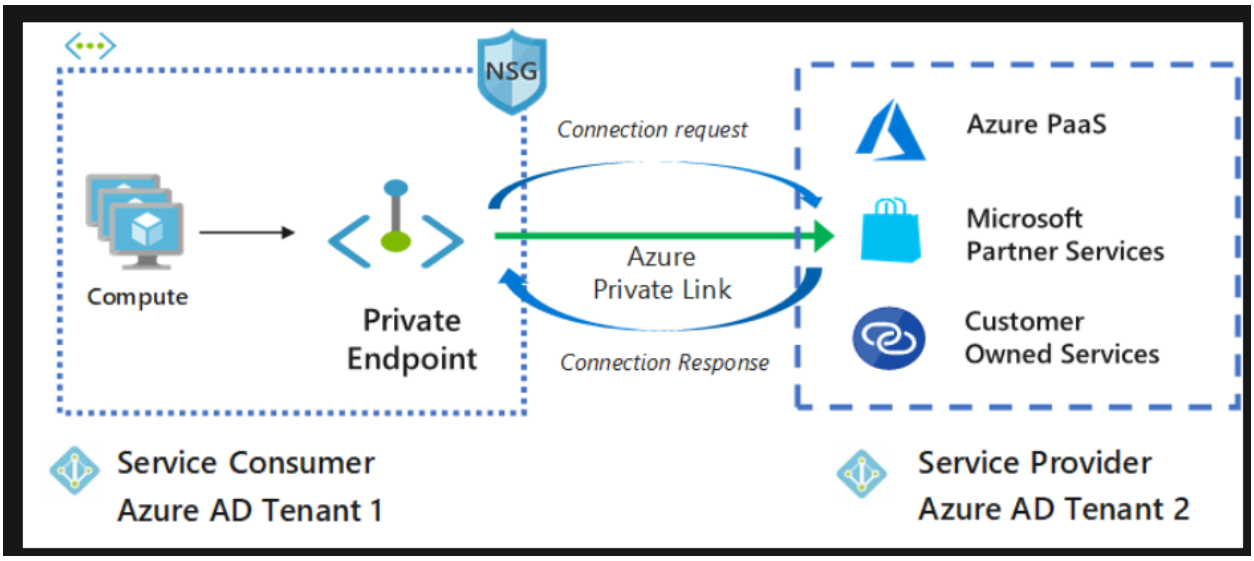
- **Endpoints always take service traffic directly from your virtual network to the service on the Microsoft Azure backbone network**. Keeping traffic on the Azure backbone network allows you to continue auditing and monitoring outbound Internet traffic from your virtual networks, through forced-tunneling, without impacting service traffic.
- **Simple to set up with less management overhead**. You no longer need reserved, public IP addresses in your virtual networks to secure Azure resources through IP firewall. There are no NAT or gateway devices required to set up the service endpoints. Service endpoints are configured through a simple click on a subnet. There is no additional overhead to maintaining the endpoints.

Important

With service endpoints, the source IP addresses of the virtual machines in the subnet for service traffic switches from using public IPv4 addresses to using private IPv4 addresses. Existing Azure service firewall rules using Azure public IP addresses will stop working with this switch. Please ensure Azure service firewall rules allow for this switch before setting up service endpoints. You may also experience temporary interruption to service traffic from this subnet while configuring service endpoints.

Scenarios

- **Peered, connected, or multiple virtual networks**: To secure Azure services to multiple subnets within a virtual network or across multiple virtual networks, you can enable service endpoints on each of the subnets independently, and secure Azure service resources to all of the subnets.
- **Filtering outbound traffic from a virtual network to Azure services**: If you want to inspect or filter the traffic sent to an Azure service from a virtual network, you can deploy a network virtual appliance within the virtual network. You can then apply service endpoints to the subnet where the network virtual appliance is deployed, and secure Azure service resources only to this subnet. This scenario might be helpful if you want use network virtual appliance filtering to restrict Azure service access from your virtual network only to specific Azure resources.
- **Securing Azure resources to services deployed directly into virtual networks**: You can directly deploy various Azure services into specific subnets in a virtual network. You can secure Azure service resources to managed service subnets by setting up a service endpoint on the managed service subnet.
- **Disk traffic from an Azure virtual machine**: Virtual Machine Disk traffic for managed and unmanaged disks isn't affected by service endpoints routing changes for Azure Storage. This traffic includes diskIO as well as mount and unmount. You can limit REST access to page blobs to select networks through service endpoints and Azure Storage network rules.

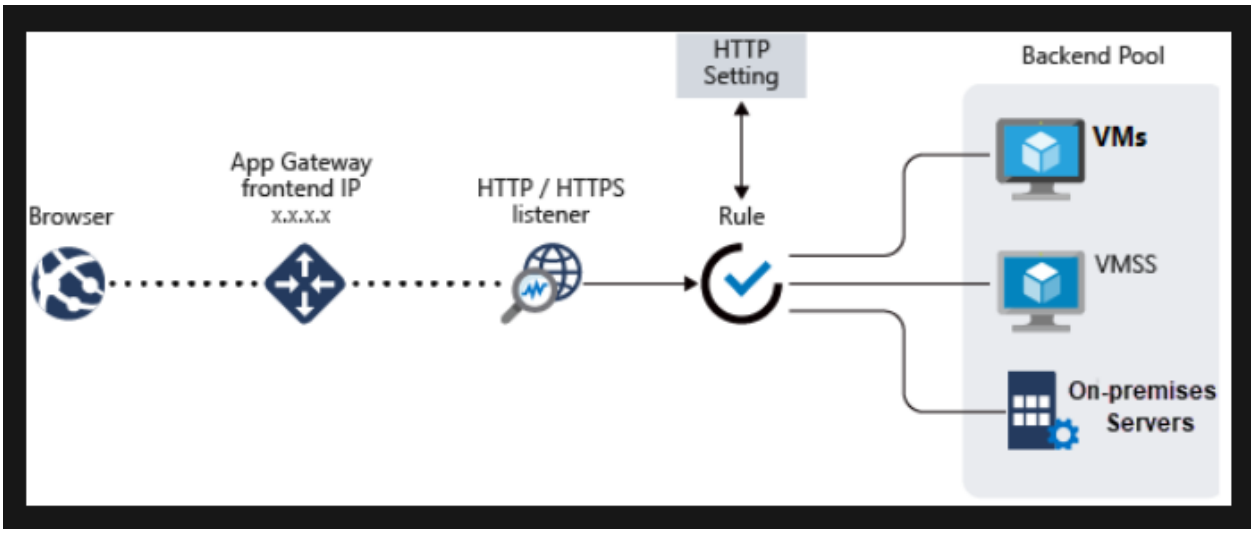


There are two connection approval methods that a Private Link service consumer can choose from:

- **Automatic**: If the service consumer has RBAC permissions on the service provider resource, the consumer can choose the automatic approval method. In this case, when the request reaches the service provider resource, no action is required from the service provider and the connection is automatically approved.
- **Manual**: On the contrary, if the service consumer doesn't have RBAC permissions on the service provider resource, the consumer can choose the manual approval method. In this case, the connection request appears on the service resources as Pending. The service provider has to manually approve the request before connections can be established. In manual cases, service consumer can also specify a message with the request to provide more context to the service provider.

The service provider has following options to choose from for all Private Endpoint connections:

- **Approved**
- **Reject**
- **Remove**



Application Gateway includes the following features:

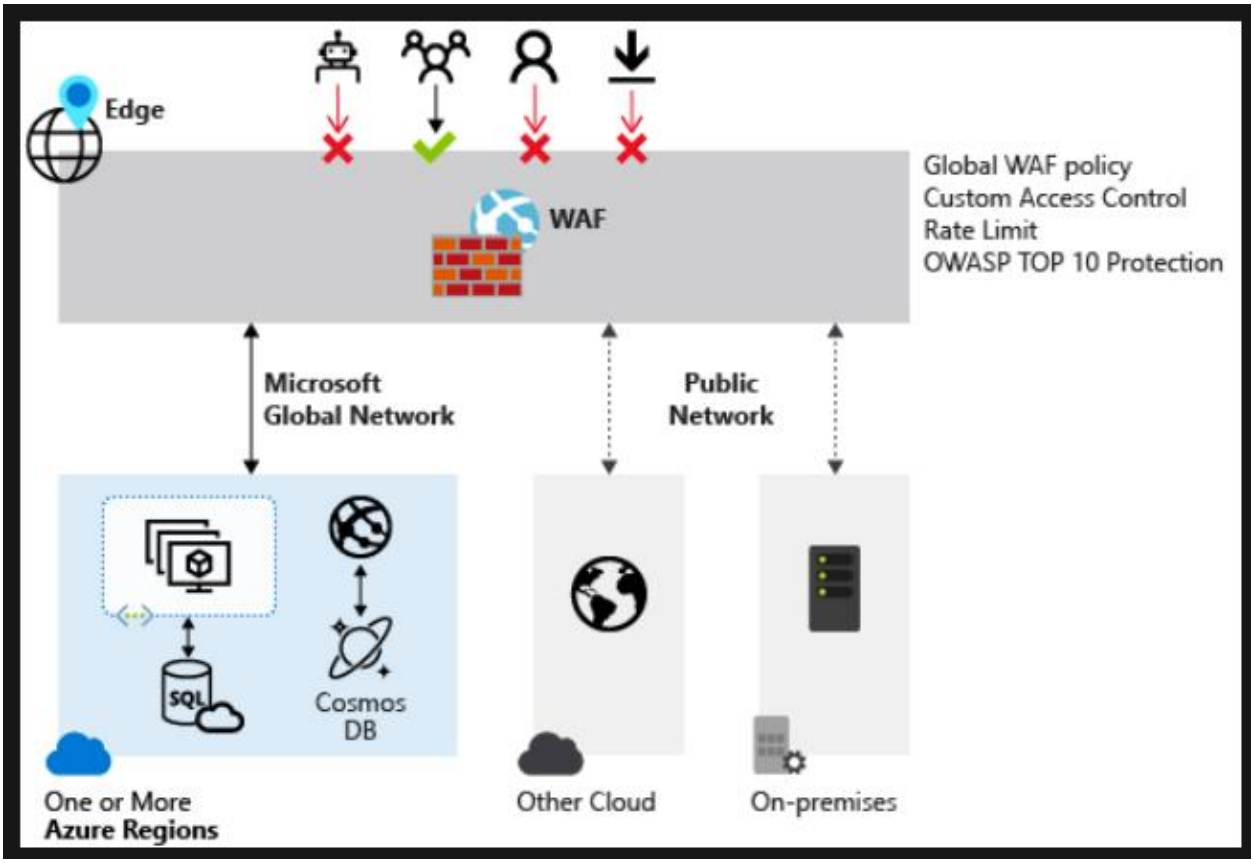
- **Secure Sockets Layer (SSL/TLS) termination** - Application gateway supports SSL/TLS termination at the gateway, after which traffic typically flows unencrypted to the backend servers. This feature allows web servers to be unburdened from costly encryption and decryption overhead.
- **Autoscaling** - Application Gateway Standard_v2 supports autoscaling and can scale up or down based on changing traffic load patterns. Autoscaling also removes the requirement to choose a deployment size or instance count during provisioning.
- **Zone redundancy** - A Standard_v2 Application Gateway can span multiple Availability Zones, offering better fault resiliency and removing the need to provision separate Application Gateways in each zone.
- **Static VIP** - The application gateway Standard_v2 SKU supports static VIP type exclusively. This ensures that the VIP associated with application gateway doesn't change even over the lifetime of the Application Gateway.
- **Web Application Firewall** - Web Application Firewall (WAF) is a service that provides centralized protection of your web applications from common exploits and vulnerabilities. WAF is based on rules from the OWASP (Open Web Application Security Project) core rule sets 3.1 (WAF_v2 only), 3.0, and 2.2.9.
- **Ingress Controller for AKS** - Application Gateway Ingress Controller (AGIC) allows you to use Application Gateway as the ingress for an Azure Kubernetes Service (AKS) cluster.
- **URL-based routing** - URL Path Based Routing allows you to route traffic to back-end server pools based on URL Paths of the request. One of the scenarios is to route requests for different content types to different pool.
- **Multiple-site hosting** - Multiple-site hosting enables you to configure more than one web site on the same application gateway instance. This feature allows you to configure a more efficient

topology for your deployments by adding up to 100 web sites to one Application Gateway (for optimal performance).

- **Redirection** - A common scenario for many web applications is to support automatic HTTP to HTTPS redirection to ensure all communication between an application and its users occurs over an encrypted path.
- **Session affinity** - The cookie-based session affinity feature is useful when you want to keep a user session on the same server.
- **Websocket and HTTP/2 traffic** - Application Gateway provides native support for the WebSocket and HTTP/2 protocols. There's no user-configurable setting to selectively enable or disable WebSocket support.
- **Connection draining** - Connection draining helps you achieve graceful removal of backend pool members during planned service updates.
- **Custom error pages** - Application Gateway allows you to create custom error pages instead of displaying default error pages. You can use your own branding and layout using a custom error page.
- **Rewrite HTTP headers** - HTTP headers allow the client and server to pass additional information with the request or the response.
- **Sizing** - Application Gateway Standard_v2 can be configured for autoscaling or fixed size deployments. This SKU doesn't offer different instance sizes.

New Application Gateway v1 SKU deployments can take up to 20 minutes to provision. Changes to instance size or count aren't disruptive, and the gateway remains active during this time.

Most deployments that use the v2 SKU take around 6 minutes to provision. However, it can take longer depending on the type of deployment. For example, deployments across multiple Availability Zones with many instances can take more than 6 minutes.



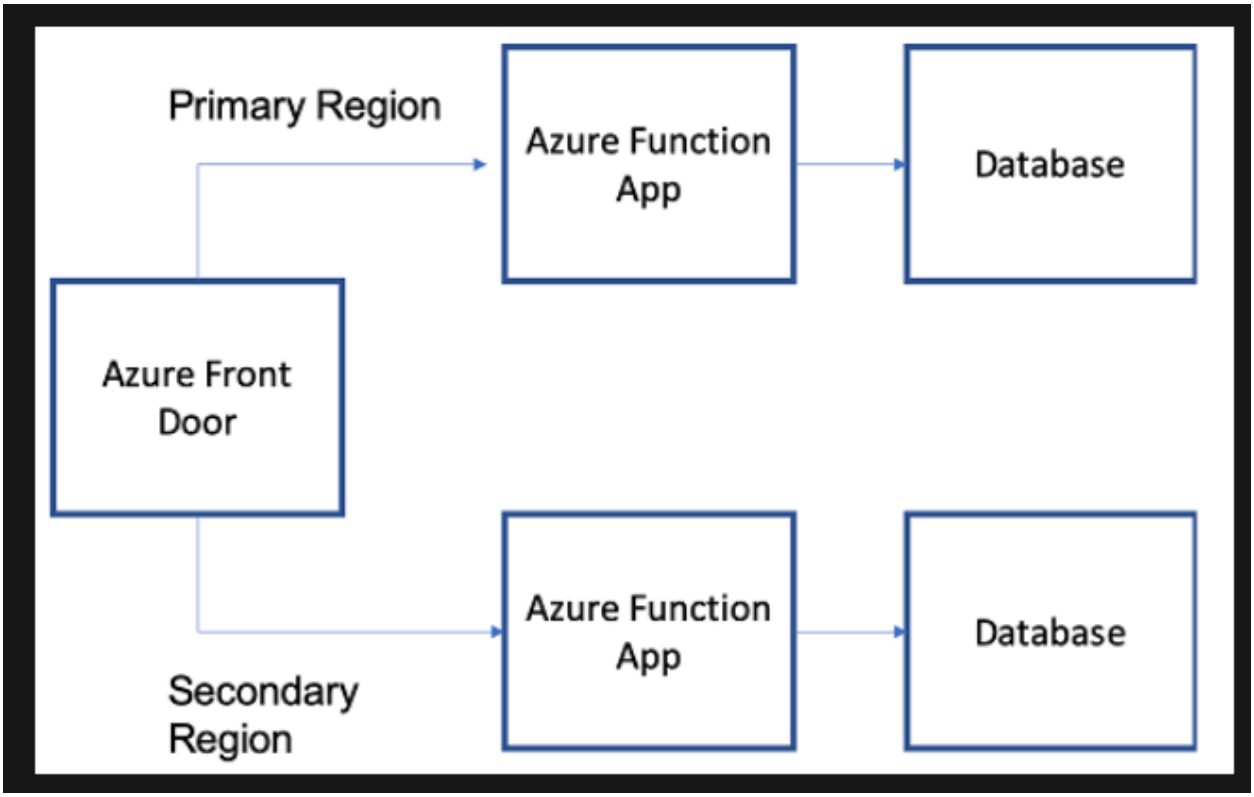
WAF

A centralized web application firewall helps make security management much simpler. A WAF also gives application administrators better assurance of protection against threats and intrusions.

A WAF solution can react to a security threat faster by centrally patching a known vulnerability, instead of securing each individual web application.

Supported service

WAF can be deployed with Azure Application Gateway, Azure Front Door, and Azure Content Delivery Network (CDN) service from Microsoft. WAF on Azure CDN is currently under public preview. WAF has features that are customized for each specific service.



Configure and manage Azure front door

Azure Front Door enables you to define, manage, and monitor the global routing for your web traffic by optimizing for best performance and instant global failover for high availability. With Front Door, you can transform your global (multi-region) consumer and enterprise applications into robust, high-performance personalized modern applications, APIs, and content that reaches a global audience with Azure.

Front Door works at Layer 7 or HTTP/HTTPS layer and uses **split TCP-based anycast protocol**. Front Door ensures that your end users promptly connect to the nearest Front Door POP (Point of Presence). So, per your routing method selection in the configuration, you can ensure that Front Door is routing your client requests to the fastest and most available application backend. An application backend is any Internet-facing service hosted inside or outside of Azure. Front Door provides a range of traffic-routing methods and backend health monitoring options to suit different application needs and automatic failover models. Like Traffic Manager, Front Door is resilient to failures, including the failure of an entire Azure region.

The following features are included with Front Door:

- **Accelerate application performance** - Using split TCP-based anycast protocol, Front Door ensures that your end users promptly connect to the nearest Front Door POP (Point of Presence).

- **Increase application availability with smart health probes** - Front Door delivers high availability for your critical applications using its smart health probes, monitoring your backends for both latency and availability and providing instant automatic failover when a backend goes down.
- **URL-based routing** - URL Path Based Routing allows you to route traffic to backend pools based on URL paths of the request. One of the scenarios is to route requests for different content types to different backend pools.
- **Multiple-site hosting** - Multiple-site hosting enables you to configure more than one web site on the same Front Door configuration.
- **Session affinity** - The cookie-based session affinity feature is useful when you want to keep a user session on the same application backend.
- **TLS termination** - Front Door supports TLS termination at the edge that is, individual users can set up a TLS connection with Front Door environments instead of establishing it over long haul connections with the application backend.
- **Custom domains and certificate management** - When you use Front Door to deliver content, a custom domain is necessary if you would like your own domain name to be visible in your Front Door URL.
- **Application layer security** - Azure Front Door allows you to author custom Web Application Firewall (WAF) rules for access control to protect your HTTP/HTTPS workload from exploitation based on client IP addresses, country code, and http parameters.
- **URL redirection** - With the strong industry push on supporting only secure communication, web applications are expected to automatically redirect any HTTP traffic to HTTPS.
- **URL rewrite** - Front Door supports URL rewrite by allowing you to configure an optional Custom Forwarding Path to use when constructing the request to forward to the backend.
- **Protocol support - IPv6 and HTTP/2 traffic** - Azure Front Door natively supports end-to-end IPv6 connectivity and HTTP/2 protocol.

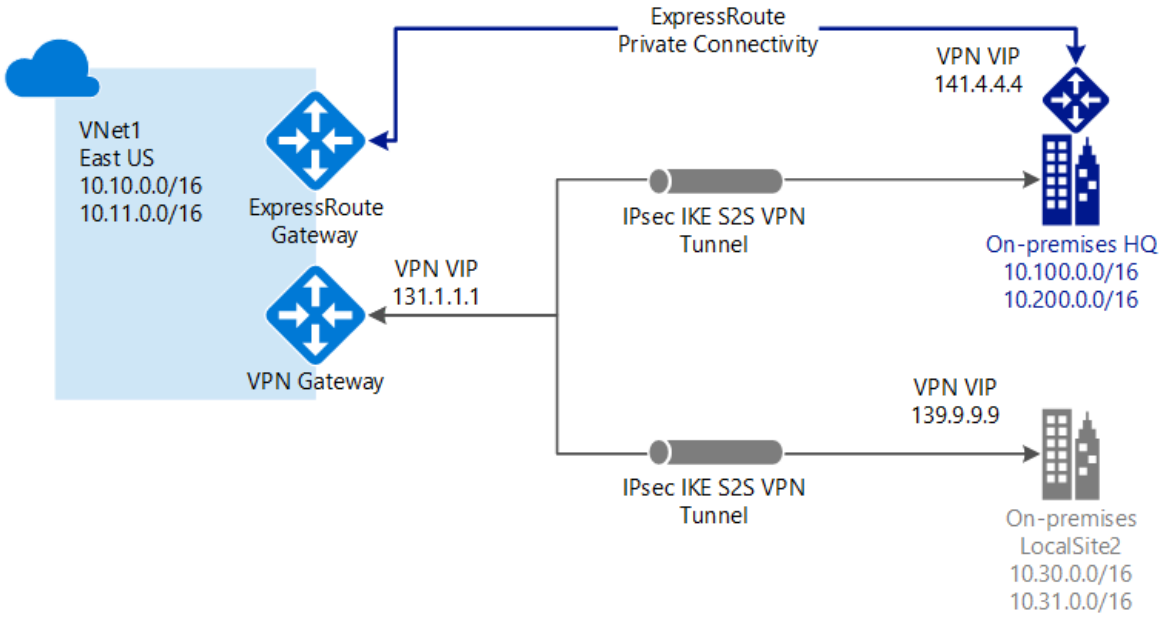
As mentioned above, routing to the Azure Front Door environments leverages Anycast for both DNS (Domain Name System) and HTTP (Hypertext Transfer Protocol) traffic, so user traffic will go to the closest environment in terms of network topology (fewest hops). This architecture typically offers better round-trip times for end users (maximizing the benefits of Split TCP). Front Door organizes its environments into primary and fallback "rings". The outer ring has environments that are closer to users, offering lower latencies. The inner ring has environments that can handle the failover for the outer ring environment in case an issue happens. The outer ring is the preferred target for all traffic, but the inner ring is necessary to handle traffic overflow from the outer ring. In terms of VIPs (Virtual Internet Protocol addresses), each frontend host, or domain served by Front Door is assigned a primary VIP, which is announced by environments in both the inner and outer ring, as well as a fallback VIP, which is only announced by environments in the inner ring.

This overall strategy ensures that requests from your end users always reach the closest Front Door environment and that even if the preferred Front Door environment is unhealthy then traffic automatically moves to the next closest environment.

Review ExpressRoute

ExpressRoute is a direct, private connection from your WAN (not over the public Internet) to Microsoft Services, including Azure. Site-to-Site VPN traffic travels encrypted over the public Internet. Being able to configure Site-to-Site VPN and ExpressRoute connections for the same virtual network has several advantages.

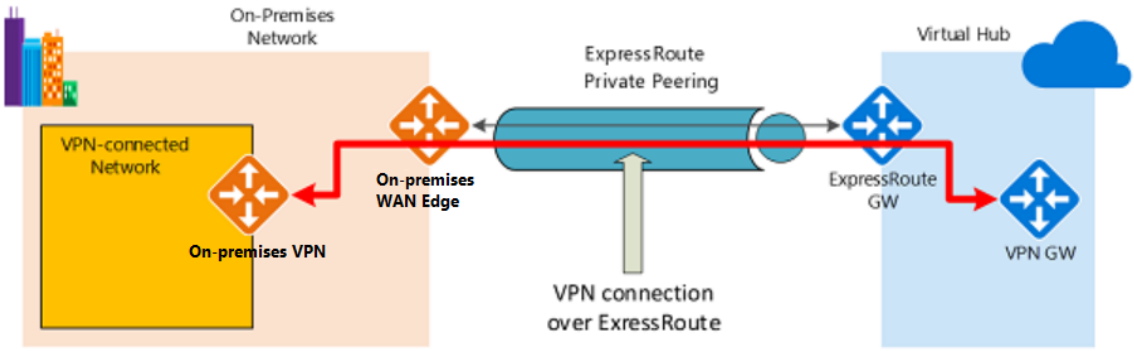
You can configure a Site-to-Site VPN as a secure failover path for ExpressRoute, or use Site-to-Site VPNs to connect to sites that are not part of your network, but that are connected through ExpressRoute. Notice that this configuration requires two virtual network gateways for the same virtual network, one using the gateway type 'Vpn', and the other using the gateway type 'ExpressRoute'.



ExpressRoute encryption

IPsec over ExpressRoute for Virtual WAN

Azure Virtual WAN uses an Internet Protocol Security (IPsec) Internet Key Exchange (IKE) VPN connection from your on-premises network to Azure over the private peering of an Azure ExpressRoute circuit. This technique can provide an encrypted transit between the on-premises networks and Azure virtual networks over ExpressRoute, without going over the public internet or using public IP addresses. The following diagram shows an example of VPN connectivity over ExpressRoute private peering.



The diagram shows a network within the on-premises network connected to the Azure hub VPN gateway over ExpressRoute private peering. The connectivity establishment is straightforward:

1. Establish ExpressRoute connectivity with an ExpressRoute circuit and private peering.
2. Establish the VPN connectivity.

An important aspect of this configuration is routing between the on-premises networks and Azure over both the ExpressRoute and VPN paths.

ExpressRoute supports a couple of encryption technologies to ensure confidentiality and integrity of the data traversing between your network and Microsoft's network.

Point-to-point encryption by MACsec

MACsec is an IEEE standard. It encrypts data at the Media Access control (MAC) level or Network Layer 2. You can use MACsec to encrypt the physical links between your network devices and Microsoft's network devices when you connect to Microsoft via ExpressRoute Direct. MACsec is disabled on ExpressRoute Direct ports by default. You bring your own MACsec key for encryption and store it in Azure Key Vault. You decide when to rotate the key.

End-to-end encryption by IPsec and MACsec

IPsec is an IETF standard. It encrypts data at the Internet Protocol (IP) level or Network Layer 3. You can use IPsec to encrypt an end-to-end connection between your on-premises network and your virtual network (VNET) on Azure.

MACsec secures the physical connections between you and Microsoft. IPsec secures the end-to-end connection between you and your virtual networks on Azure. You can enable them independently.

ExpressRoute Direct

ExpressRoute Direct gives you the ability to connect directly into Microsoft's global network at peering locations strategically distributed across the world. ExpressRoute Direct provides dual 100 Gbps or 10 Gbps connectivity, which supports Active/Active connectivity at scale

Key features that ExpressRoute Direct provides include, but aren't limited to:

- Massive Data Ingestion into services like Storage and Cosmos DB

- Physical isolation for industries that are regulated and require dedicated and isolated connectivity like: Banking, Government, and Retail
- Granular control of circuit distribution based on business unit

ExpressRoute Direct supports massive data ingestion scenarios into Azure storage and other big data services. ExpressRoute circuits on 100 Gbps ExpressRoute Direct now also support 40 Gbps and 100 Gbps circuit SKUs. The physical port pairs are 100 or 10 Gbps only and can have multiple virtual circuits.

ExpressRoute Direct supports both QinQ and Dot1Q VLAN tagging.

- **QinQ VLAN Tagging** allows for isolated routing domains on a per ExpressRoute circuit basis. Azure dynamically allocates an S-Tag at circuit creation and cannot be changed. Each peering on the circuit (Private and Microsoft) will utilize a unique C-Tag as the VLAN. The C-Tag is not required to be unique across circuits on the ExpressRoute Direct ports.
- **Dot1Q VLAN Tagging** allows for a single tagged VLAN on a per ExpressRoute Direct port pair basis. A C-Tag used on a peering must be unique across all circuits and peerings on the ExpressRoute Direct port pair.

ExpressRoute Direct provides the same enterprise-grade SLA with Active/Active redundant connections into the Microsoft Global Network. ExpressRoute infrastructure is redundant and connectivity into the Microsoft Global Network is redundant and diverse and scales accordingly with customer requirements.