Microsoft Azure

WorkshopPLUS- Networking Essentials

Module # 2
Deliver

CONNECT & EXTEND

Virtual Networks

Routing/NSGs

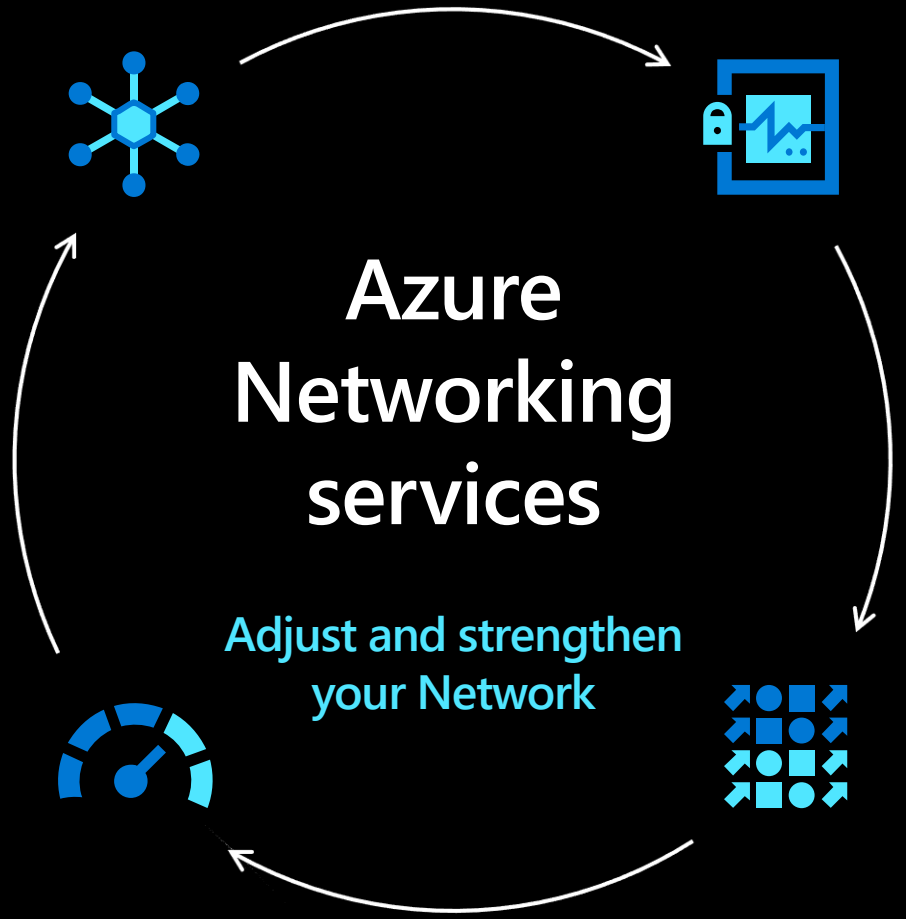
ExpressRoute/VPN

Virtual WAN

MONITOR

Azure Monitor

Azure Network Watcher



PROTECT

Bastion

Azure Firewall

DDoS Protection

Web Application Firewall

PrivateLink/Private Endpoints

DELIVER

DNS, Azure Load Balancer

Traffic Manager

Application Gateway

CDN/Azure Front Door

Azure DNS

Azure DNS

Azure DNS

- Manage DNS seamlessly with your Azure services
- Azure DNS uses anycast networking
- Globally distributed architecture, resilient to multiple region failure
- 99.99% Availability SLA

All common DNS record types

- A, AAAA, CNAME, MX, NS, PTR, SOA, SRV, and TXT

Flexible DNS management

- Integrated with Azure Resource Manager for role-based access control, tagging, and template-based deployment—for both zones and record sets
- Azure Portal, PowerShell, and CLI
- REST API and SDKs for application integration

Name resolution

DNS for public zones

- CNAMES can be created in customer zones
- Support for reverse DNS

DNS for private zones

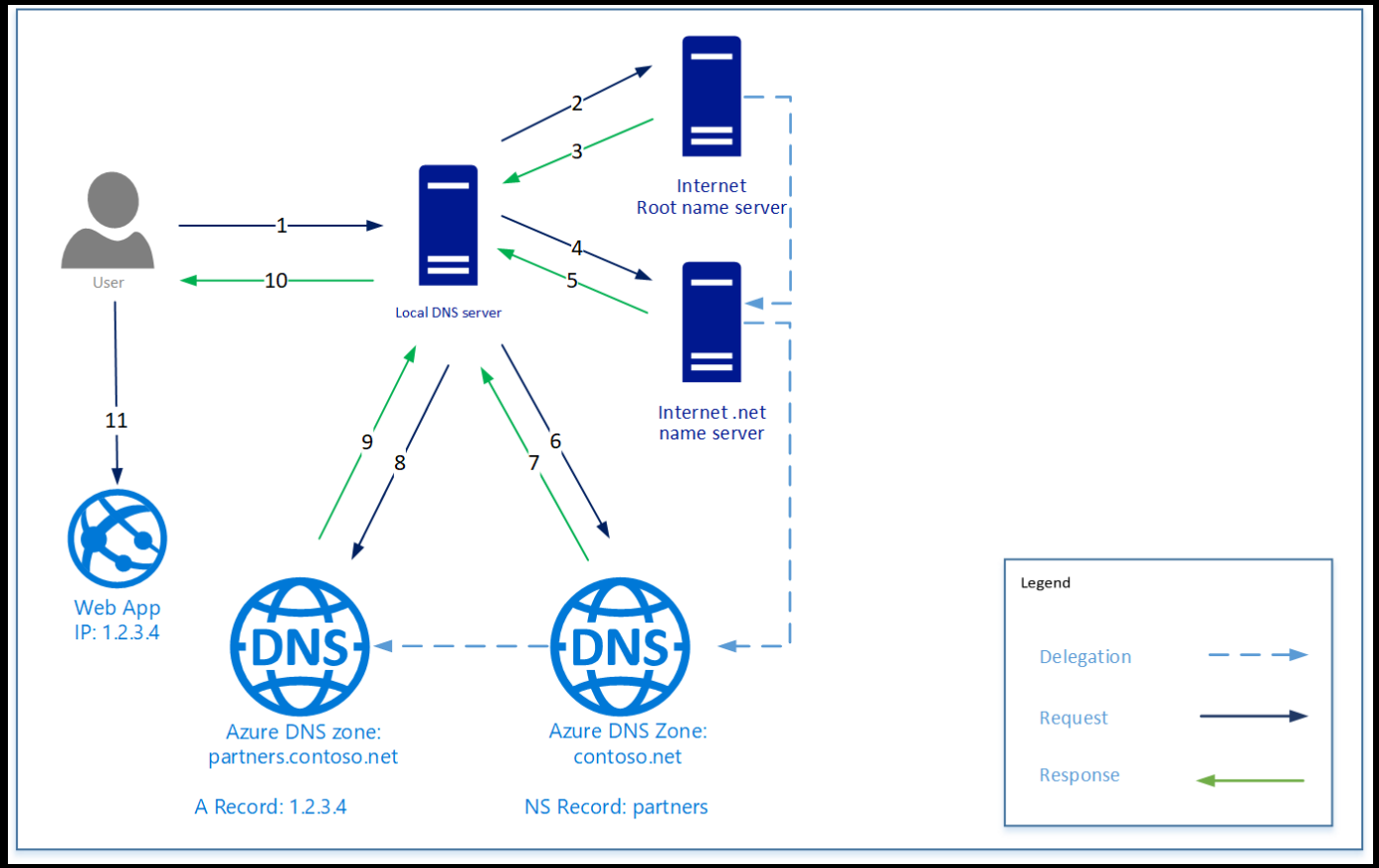
- Vnet link zones to allow direct queries
- Assign DNS servers per VNET
- Automatic registration to defined Azure zones

DNS hosting

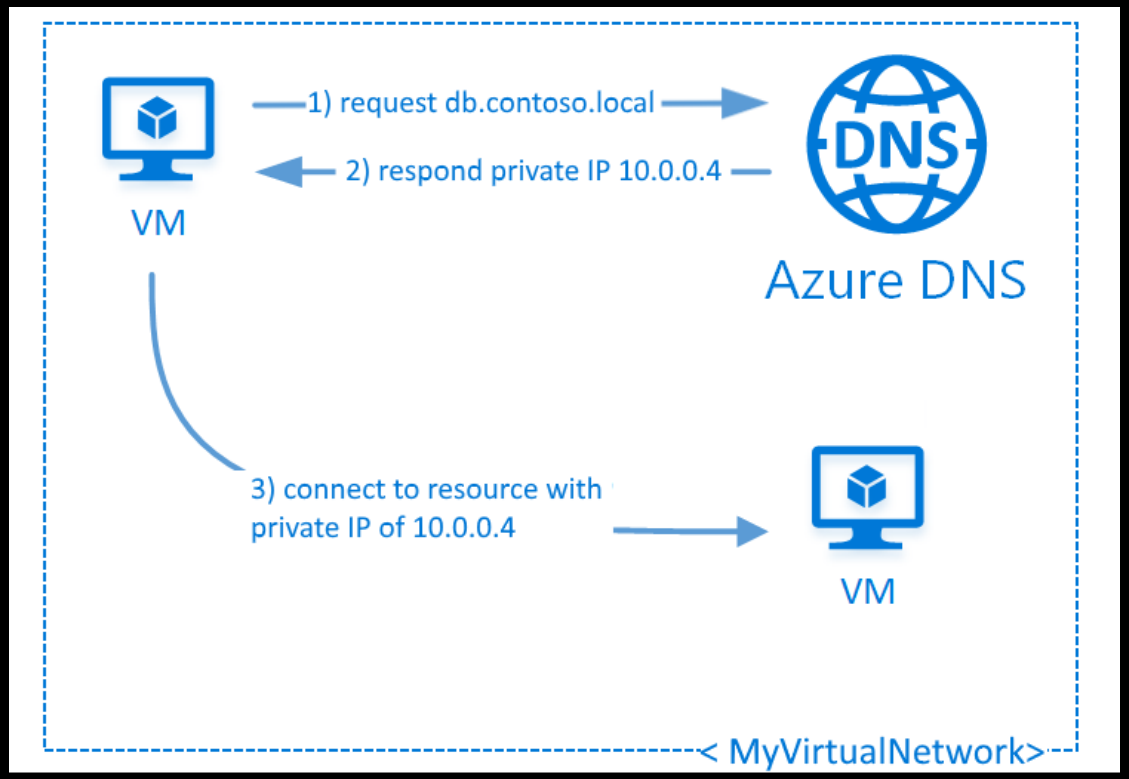
- Azure hosted public-facing DNS servers
- Support zone names with split-horizon views

Azure DNS Public Zones

- In order to configure name servers, you need to own the domain to set the correct name servers for the domain name with the registrar.
- Azure DNS supports all common DNS record types: A, AAAA, CAA, CNAME, MX, NS, PTR, SOA, SRV, and TXT
- *Authoritative* DNS service that hosts DNS zones. It answers DNS queries for records in those zones only
- Azure DNS can be used to host your reverse lookup zones and manage the PTR records for both IPv4 and IPv6



Azure DNS Private Zones



Host DNS Zones in your private VNET

- Eliminates need to add custom DNS solution
- Enables ability to use custom domain names

Split-Horizon DNS Support

- Create zones with the same name that resolve to different answers
- Provide a dedicated version of a service for use inside a vNet

Hostname resolution between vNets

- Private DNS zones can be shared between vNets – vnet link
- Simplify cross-network and service discovery scenarios such as vNet peering
- Autoregister VMs deployed in vNets when vNet linking
- Privatelink zones to reach Azure PaaS resources privately

Azure Load Balancer

What is a load balancer?

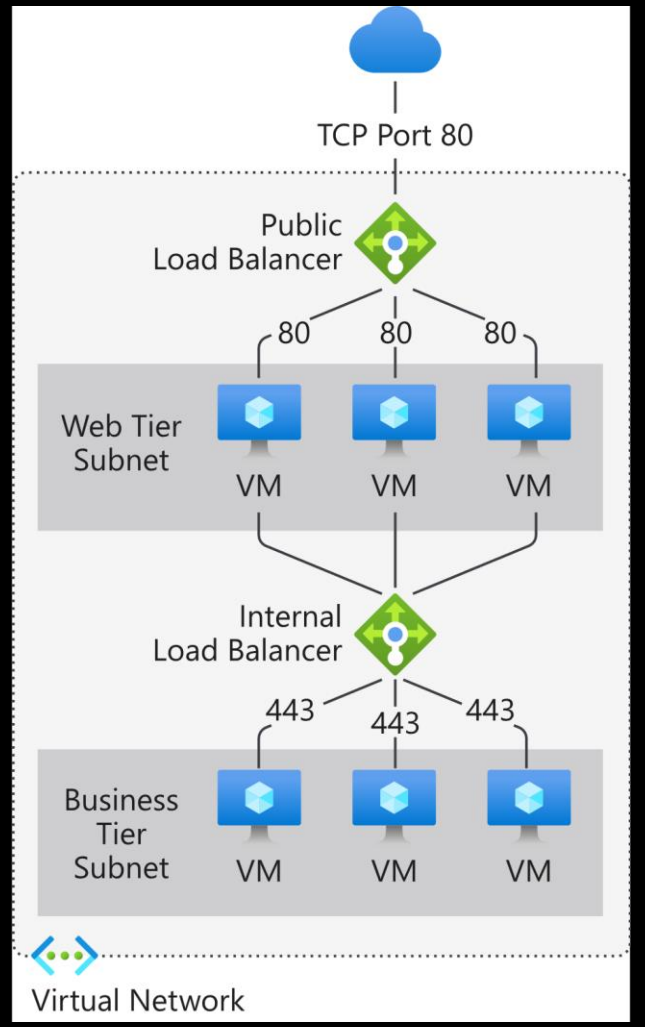
A load balancer is a layer 4 (TCP, UDP) and/or Layer 7 (Application) device that distributes incoming network traffic among a healthy cluster of servers.

That device can be physical or virtualized like Azure Load Balancer.

Load balancing improves responsiveness and increases availability of applications by checking for example if servers have enough resources, close enough to the request.

Azure Load Balancers

- Azure Load Balancer is a Layer 4 (TCP, UDP) load balancer that distributes incoming traffic among healthy instances of services defined in a load-balanced set.
- There are two types of Load Balancers:
  - **Public -** which is used to load balance incoming Internet traffic to virtual machines in a virtual network
  - **Internal -** which is used to load balance traffic between virtual machines in a virtual network, between virtual machines in cloud services, or between on-premises computers and virtual machines in a cross-premises virtual network
- Can also forward external traffic to a specific virtual machine



Azure Load Balancers

There are two SKUs of Load Balancers:

Standard

Basic

There are multiple differences between Standard and Basic SKUs, in terms of the backend pool size, outbound rules, and SLA for example.

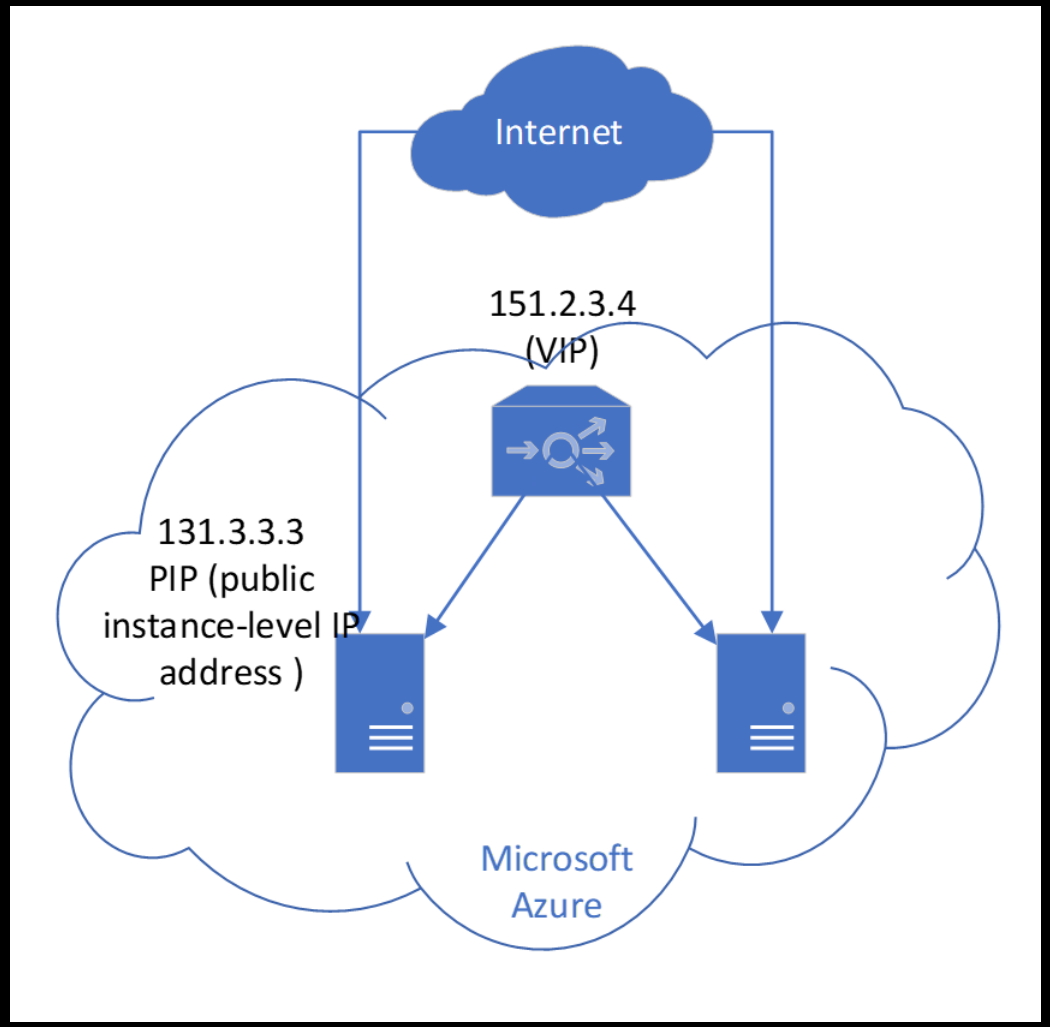
Note: You need to match SKUs for Load Balancer and Public IP resources. You can't have a mixture of Basic SKU resources and Standard SKU resources.

Azure Internal Load Balancer - ILB

Provides load balancing for machines inside of a Virtual network

- Within a virtual network
- Between virtual networks that are peered
- On-prem connected networks
- Between virtual machines in a virtual network
- Private IPs needed as the frontend, not publicly accessible

Internet IP Addresses and Azure Load Balancers



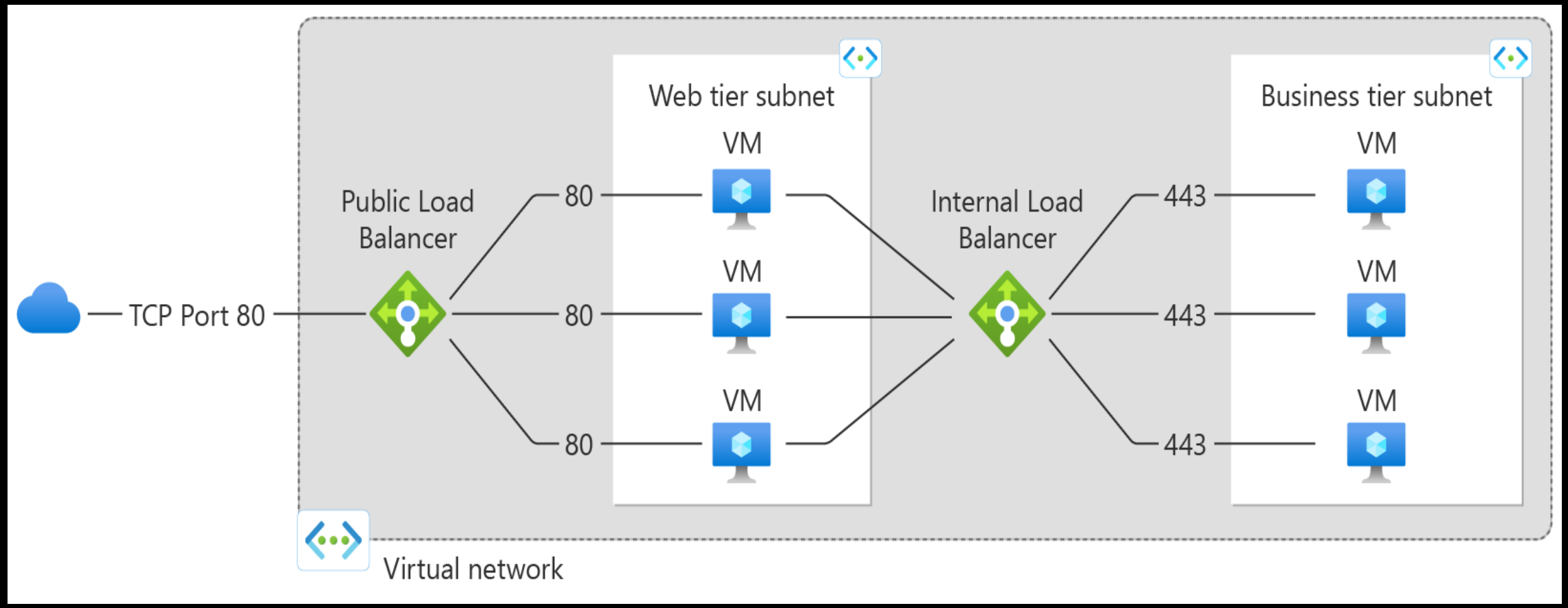
Public IP Addresses in Azure

- Can be attached for instance (VM) level access or an Azure load balancer

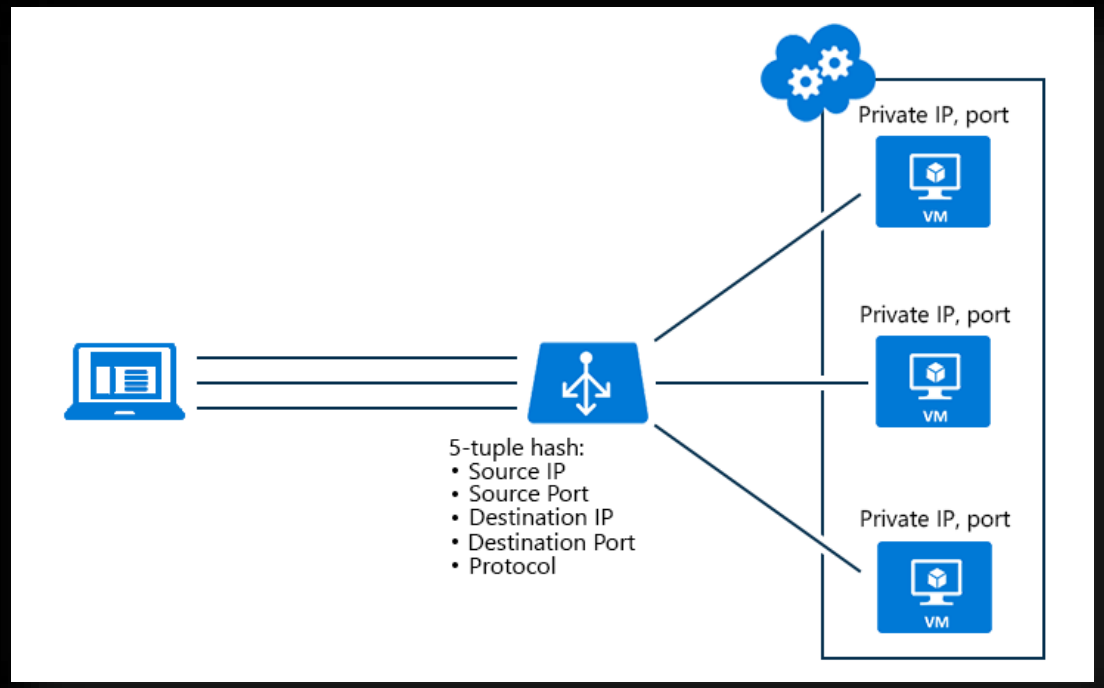
Load balanced IP (VIP)

- Internet IP load balanced among one or more VM instances
- Allows port redirection
- Primarily for load balanced, highly available, or auto-scale scenarios

Multi-tier App using both Public and Internal LB



Hash-based Distribution Mode



Default algorithm is 5-tuple hash

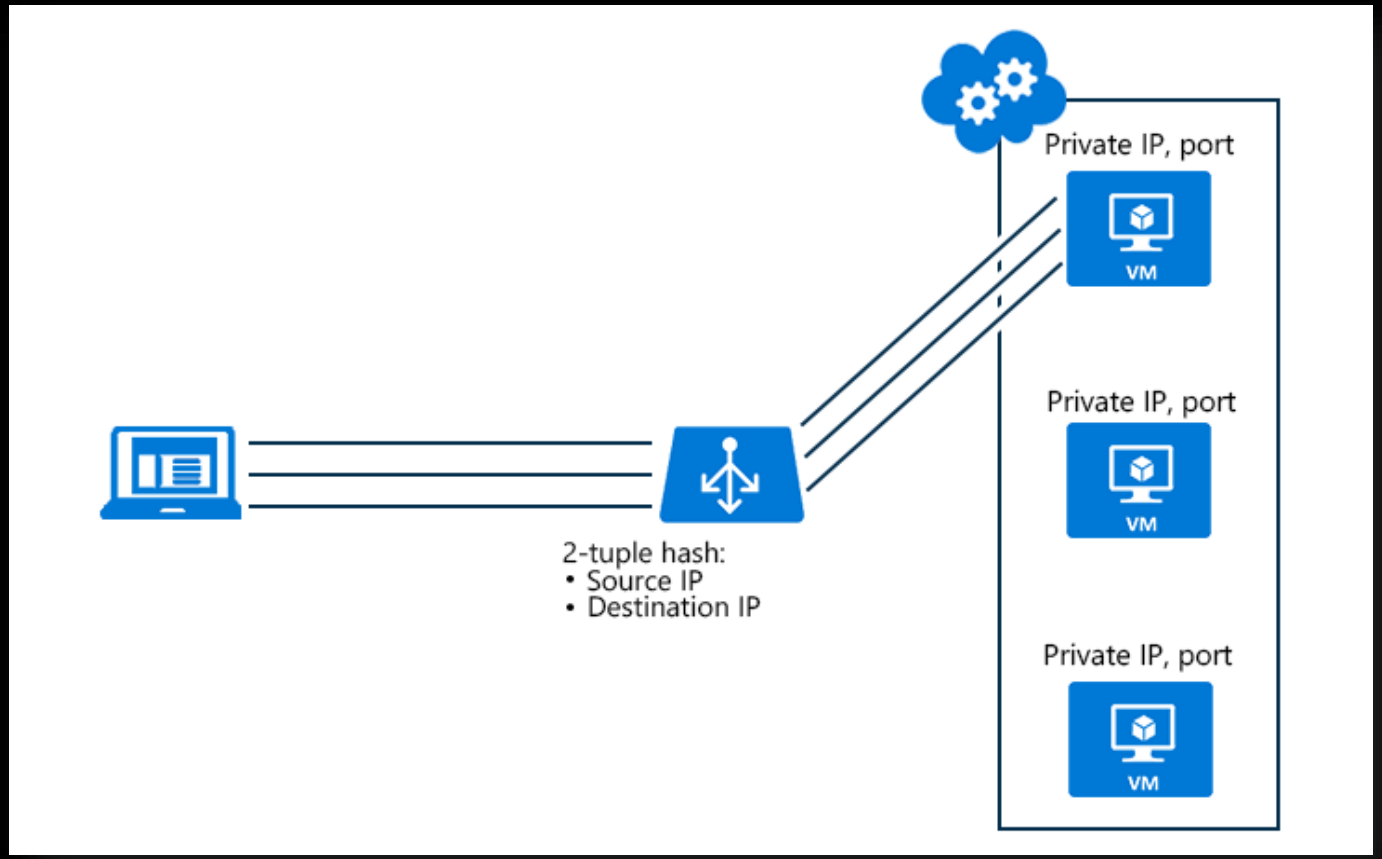
- Source IP
- Source Port
- Destination IP
- Destination Port
- Protocol

Provides stickiness only within transport session

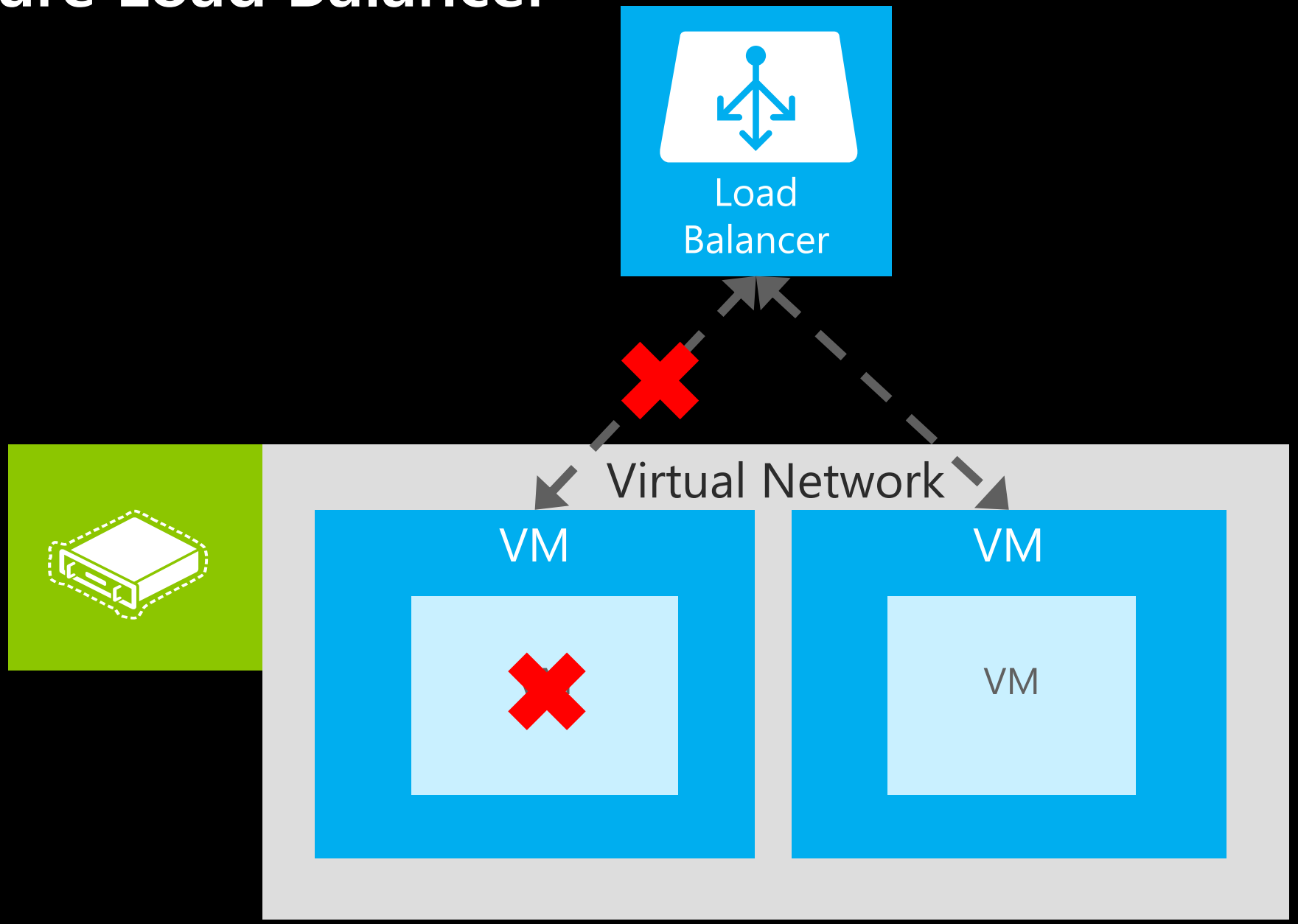
Packets in same session are directed to same DIP

Source IP Affinity

- Provide session affinity
- 2-tuple (Source IP, Destination IP)
- 3-tuple (Source IP, Destination IP, Protocol)
- Solves common affinity related problems like RD Gateway scenarios



Azure Load Balancer



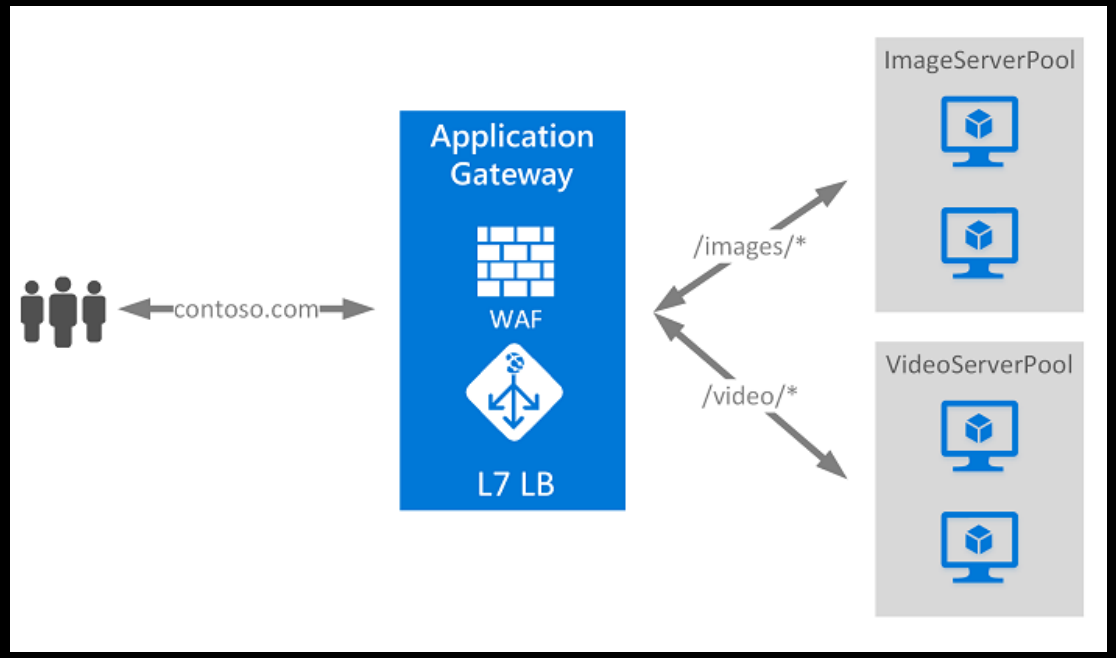
Questions?

Azure Application Gateway

What is Azure Application Gateway?

- Fully Azure managed, highly available and scalable dedicated virtual appliance
- Web traffic load balancer (Layer 7 of the OSI Model). Routing based on:
  - Source IP address and port
  - Incoming URL

- Can act as:
- Internet facing gateway
- Internal only gateway
- Combination of both



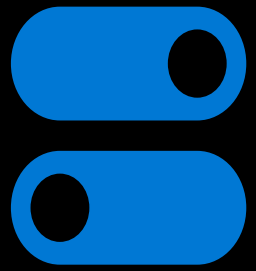
Application Gateway

Sizes (Standard):

Small

Medium

Large



Tiers:

Standard

Standard V2

WAF

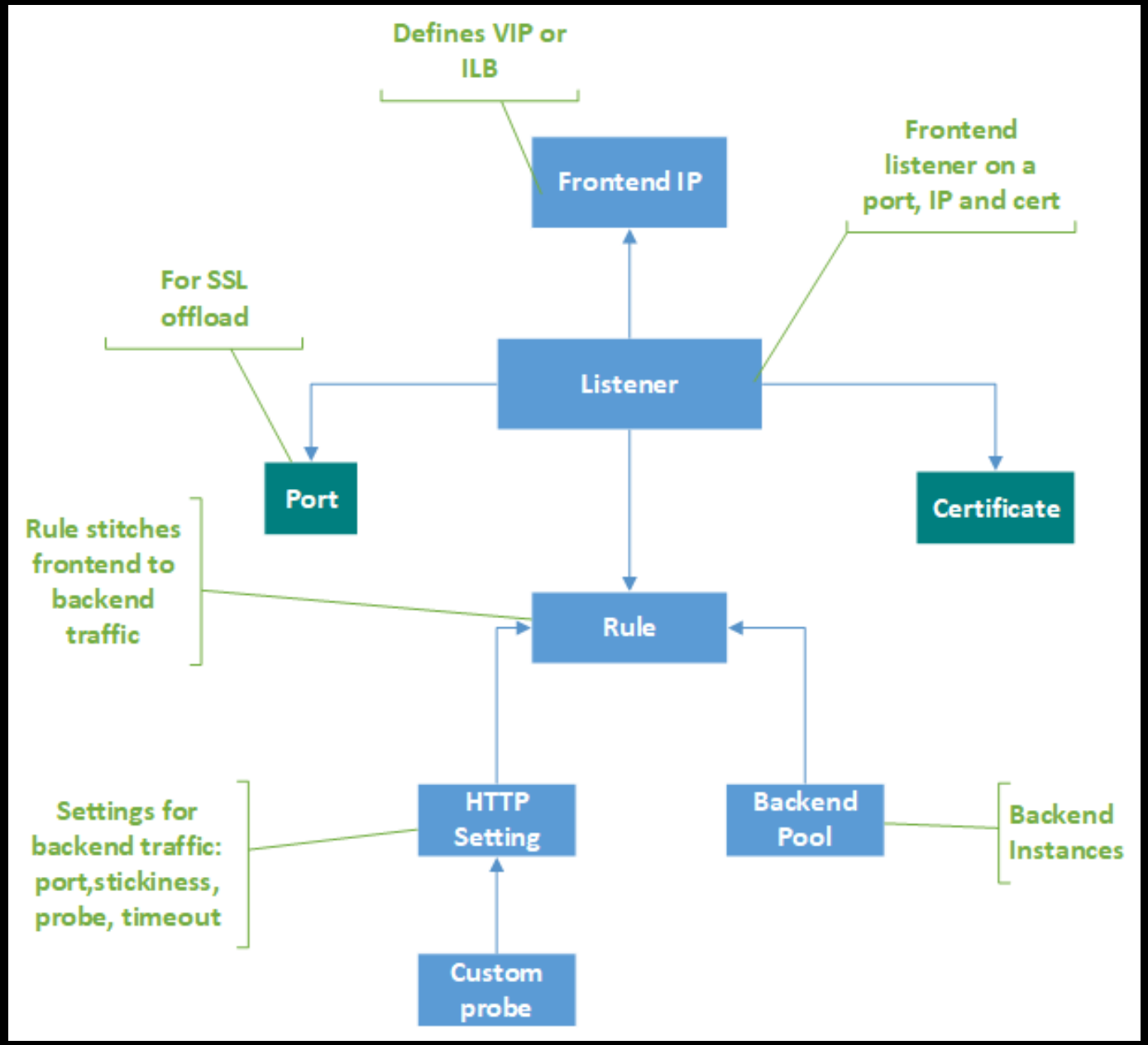
WAF V2

Application Gateway - Features

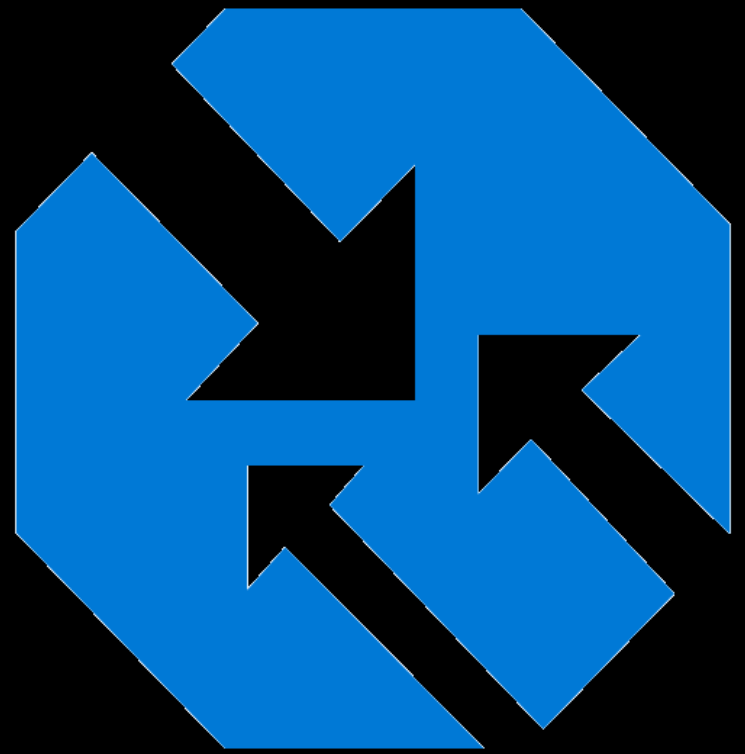
- Web Application Firewall
- HTTP load balancing
- Cookie-based session affinity
- Secure Sockets Layer (SSL) offload
- End to End SSL
- URL-based content routing
- Multi-site routing
- WebSocket support
- Health monitoring
- Advanced diagnostics

Application Gateway – Components

- Frontend IP addresses
- Listeners
- Request Routing Rules
- HTTP Settings
- Backend Pools
- Health Probes



Azure Traffic Manager



What is Azure Traffic Manager?

Azure Traffic Manager is a **DNS-based** traffic load balancer that enables you to distribute traffic optimally to services across **global** Azure regions, while providing high availability and responsiveness.

Traffic Manager Features:

- Increase Application Availability
- Improve Application Performance
- Perform Service Maintenance without Downtime
- Combine Hybrid Applications
- Distribute Traffic for Complex Deployments

How does Traffic Manager work?

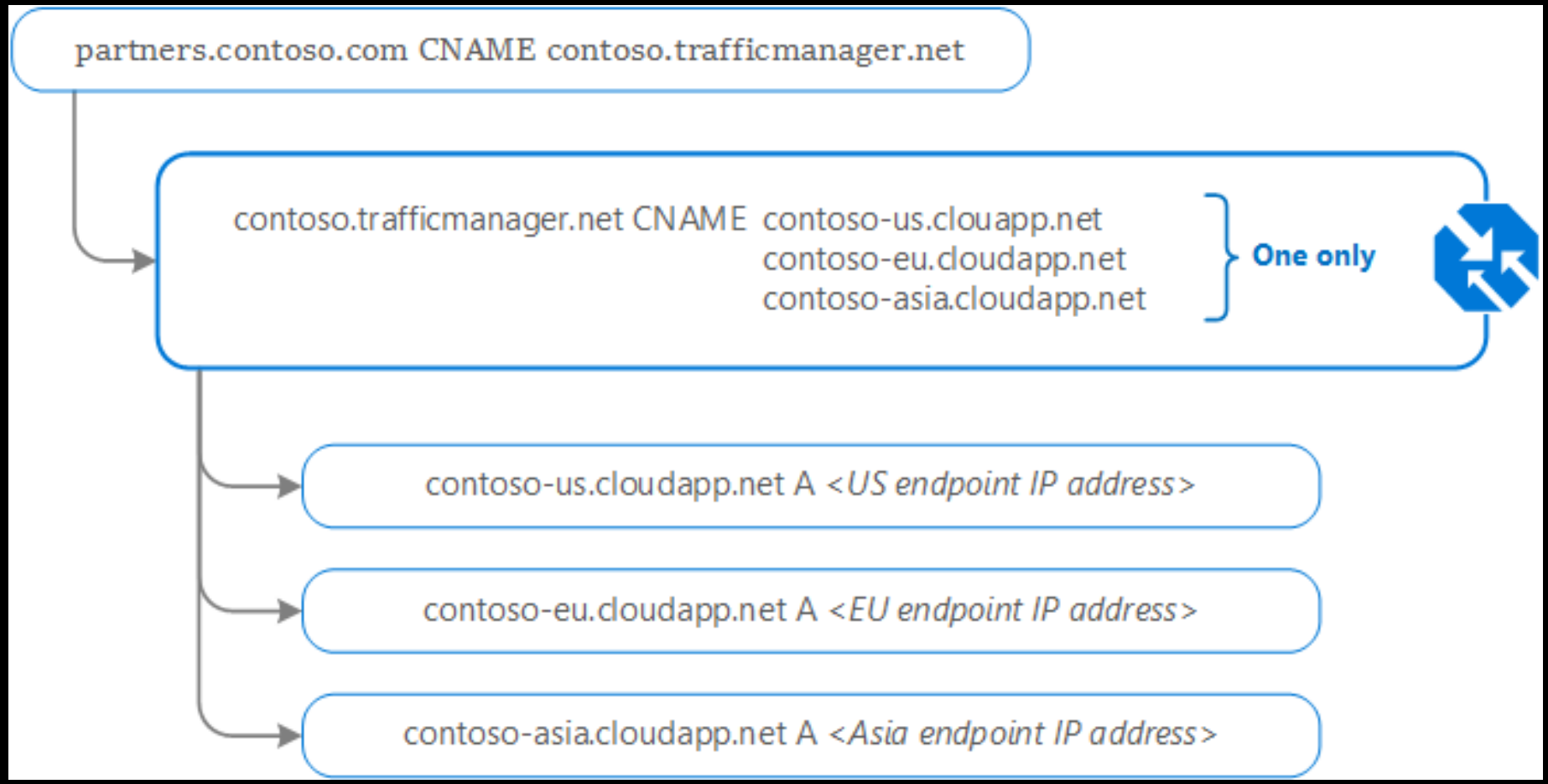
Traffic Manager provides two key benefits:

- Distribution of traffic according to one of several traffic routing methods
- Continuous monitoring of endpoint health and automatic failover when endpoints fail

When a client attempts to connect to a service, it must first resolve the DNS name of the service to an IP address. The client then connects to that IP address to access the service.

The most important point to understand is that Traffic Manager works at the **DNS** level.

Traffic Manager Example:



How Endpoint Monitoring Works

- If the monitoring protocol is set as **HTTP or HTTPS**, the Traffic Manager probing agent makes a **GET** request to the endpoint using the protocol, port, and relative path given.
- If it gets back a **200-OK** response, or any of the responses configured in the Expected status code *ranges, then that endpoint is considered healthy.
- If the response is a different value, or, if no response is received within the timeout period specified, then the Traffic Manager probing agent re-attempts according to the Tolerated Number of Failures setting (no re-attempts are done if this setting is 0).
- If the number of consecutive failures is higher than the Tolerated Number of Failures setting, then that endpoint is marked as unhealthy.

Endpoint Monitoring

- Protocol
- Port
- Path
- Custom Header Settings
- Expected Status Code Changes
- Probing Interval
- Tolerated Number of Failures
- Probe Timeout

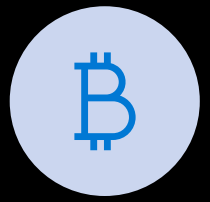
Traffic Manager Routing Methods


Priority


Weighted


Performance

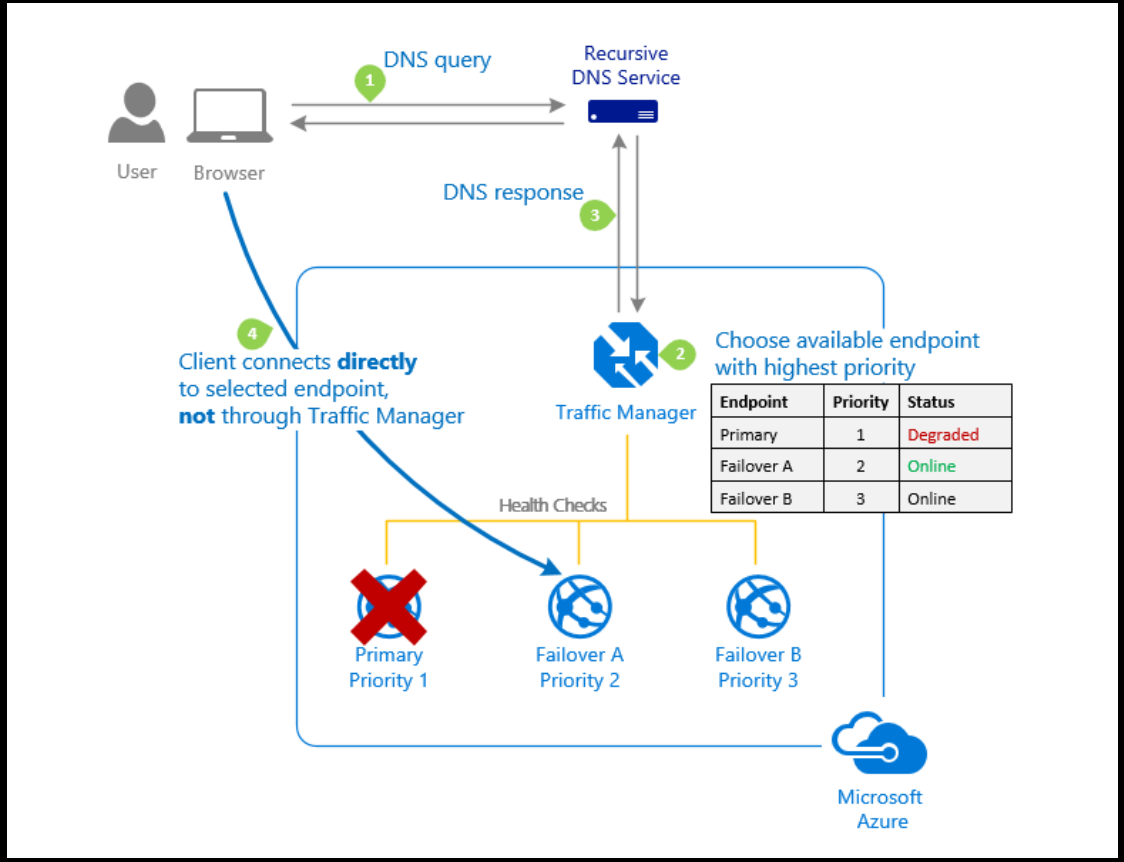

Geographic


Multi-Value


Subnet

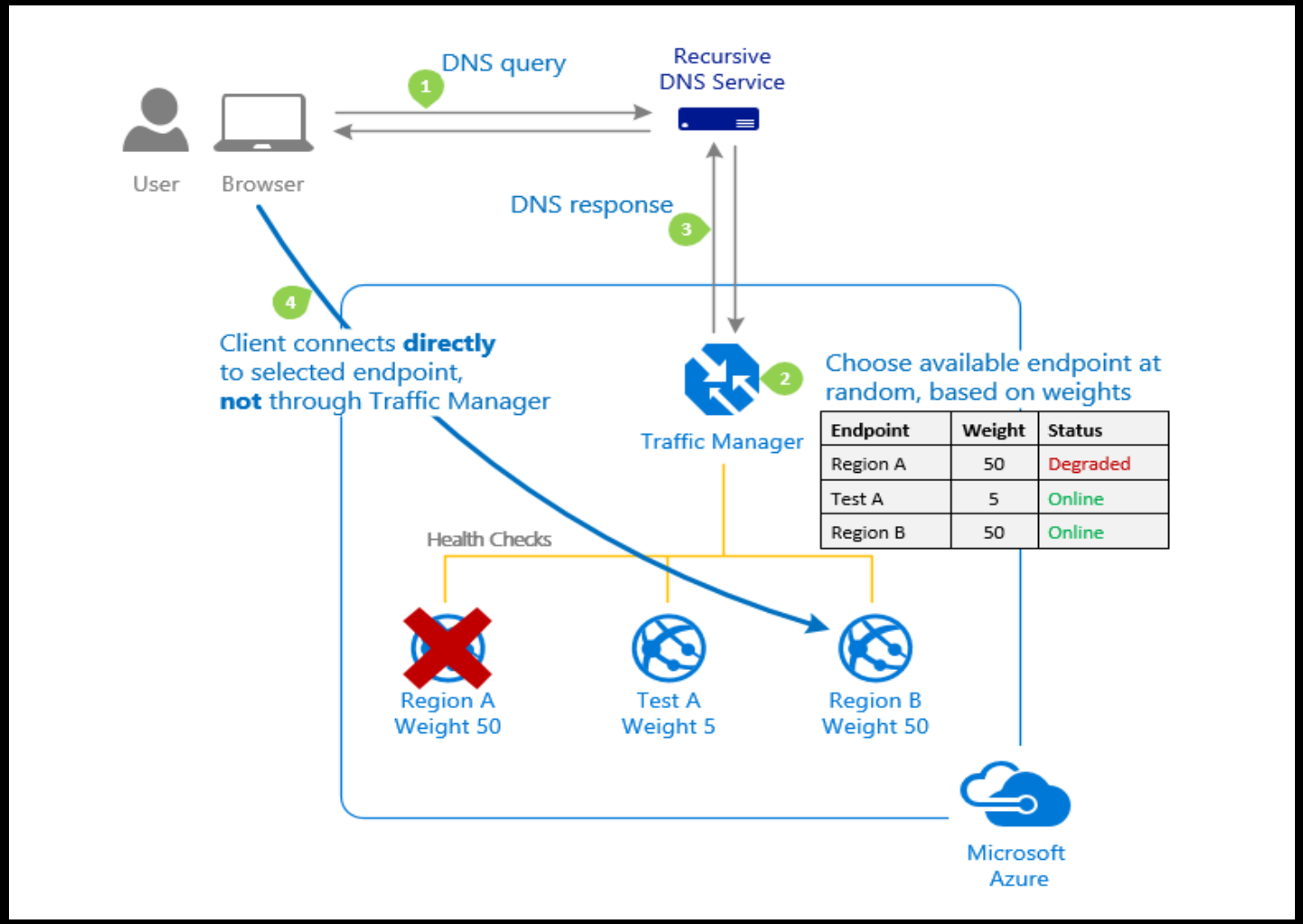
Priority traffic-routing method

Often an organization wants to provide reliability for its services by deploying one or more backup services in case their primary service goes down. The 'Priority' traffic-routing method allows Azure customers to easily implement this failover pattern.



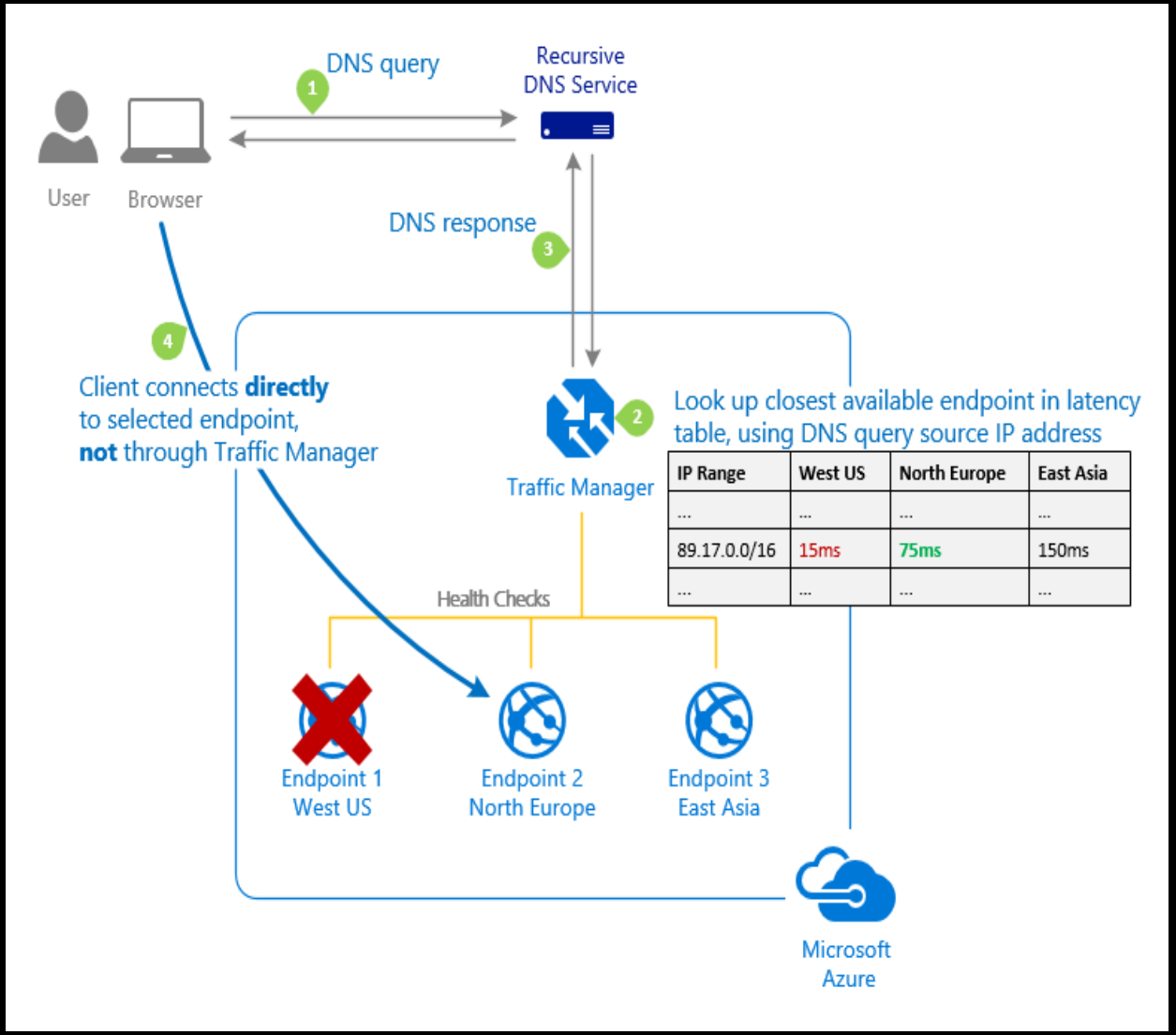
Weighted traffic-routing method

In the Weighted traffic-routing method, you assign a weight to each endpoint in the Traffic Manager profile configuration. The higher weight, the higher the priority.



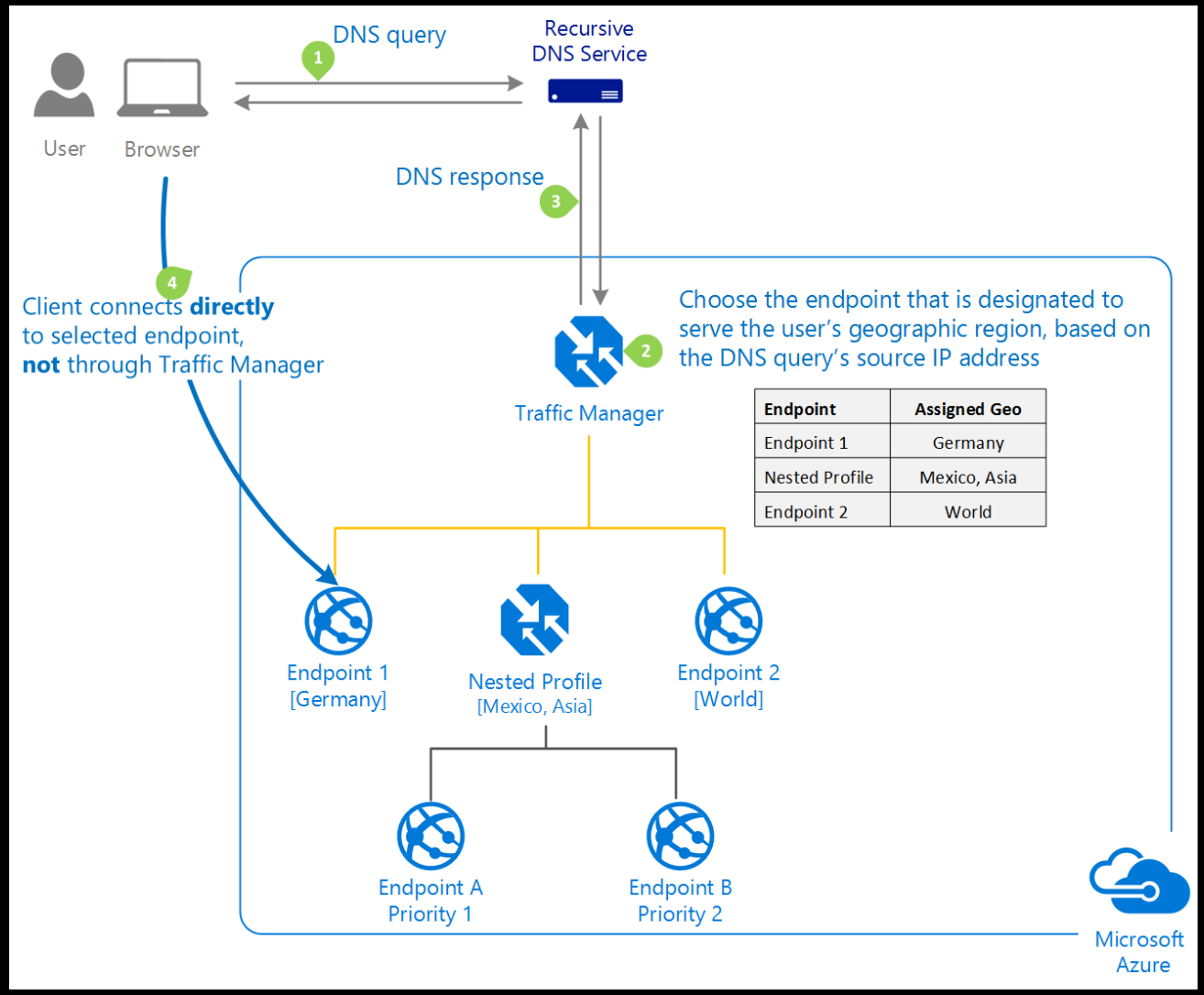
Performance traffic-routing method

Deploying endpoints in two or more locations across the globe can improve the responsiveness of many applications by routing traffic to the location that is 'closest' to you. The 'Performance' traffic-routing method provides this capability.



Geographic traffic-routing method

Traffic Manager profiles can be configured to use the Geographic routing method so that users are directed to specific endpoints (Azure, External or Nested) **based on which geographic location their DNS query originates from.**



Multi-value traffic-routing method

The Multi-value traffic-routing method allows you to get multiple healthy endpoints in a single DNS query response. This enables to caller to do client-side retries with other endpoints in the event of a returned endpoint being unresponsive.

Subnet traffic-routing method

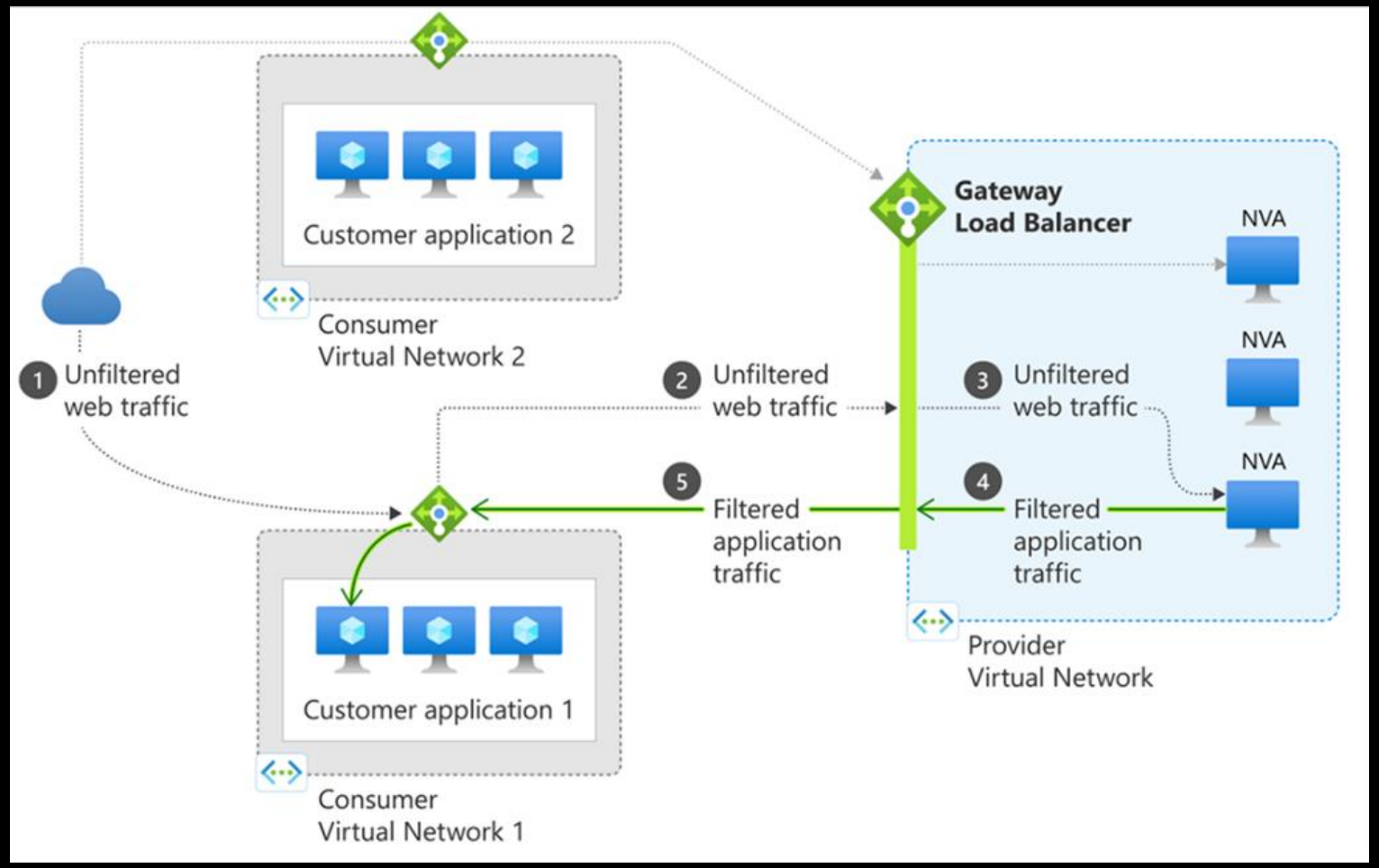
The Subnet traffic-routing method allows you to map a set of end user IP address ranges to specific endpoints in a profile. After that, if Traffic Manager receives a DNS query for that profile, it will inspect the source IP address of that request (in most cases this will be the outgoing IP address of the DNS resolver used by the caller), determine which endpoint it is mapped to and will return that endpoint in the query response.

Azure Gateway Load Balancer

What is Azure Gateway Load Balancer ?

- It provides the bump-in-the-wire technology you need to ensure all traffic to a public endpoint is first sent to the appliance before your application.
- In scenarios with NVAs, it's especially important that flows are symmetric. Gateway Load Balancer maintains flow stickiness to a specific instance in the backend pool along with flow symmetry.
- As a result, a consistent route to your network virtual appliance is ensured – without additional manual configuration. Packets traverse the same network path in both directions and appliances that need this key capability are able to function seamlessly.

How it works



GWLB can be associated to multiple consumer resources, including both *Standard Public Load Balancers and Virtual Machines with Standard Public IPs*.

When GWLB is chained to the front-end configuration or VM NIC IP configuration, unfiltered traffic from the internet will first be directed to the GWLB and then reach the configured NVAs.

The NVAs will then inspect the traffic and send the filtered traffic to the final destination, the consumer application hosted on either the load balancer or virtual machine.

Components

- Frontend IP configuration - The IP address of your Gateway Load Balancer. This IP is private only.
- Load-balancing rules - A load balancer rule is used to define how incoming traffic is distributed to all the instances within the backend pool.
- Backend pool(s) - The group of virtual machines or instances in a virtual machine scale set that is serving the incoming request.
- Tunnel interfaces - The tunnel interface enables the appliances in the backend to ensure network flows are handled as expected
- Chain - A Gateway Load Balancer can be referenced by a Standard Public Load Balancer frontend or a Standard Public IP configuration on a virtual machine. As a result, this reference is called service chaining.

Gateway Load Balancer benefits

1. Integrate virtual appliances transparently into the network path.
2. Easily add or remove network virtual appliances in the network path.
3. Scale with ease while managing costs.
4. Improve network virtual appliance availability.
5. Chain applications across regions and subscription

Limitations

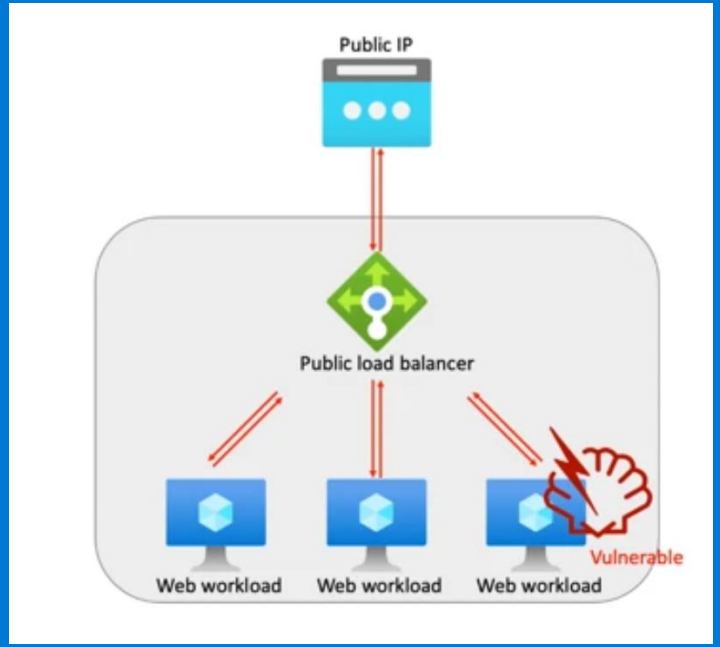
- **Gateway Load Balancer doesn't work with the Global Load Balancer tier.**
- **Cross-tenant chaining is not supported through the Azure portal.**
- **Gateway Load Balancer does not currently support IPv6**

Firewall example: User Experience

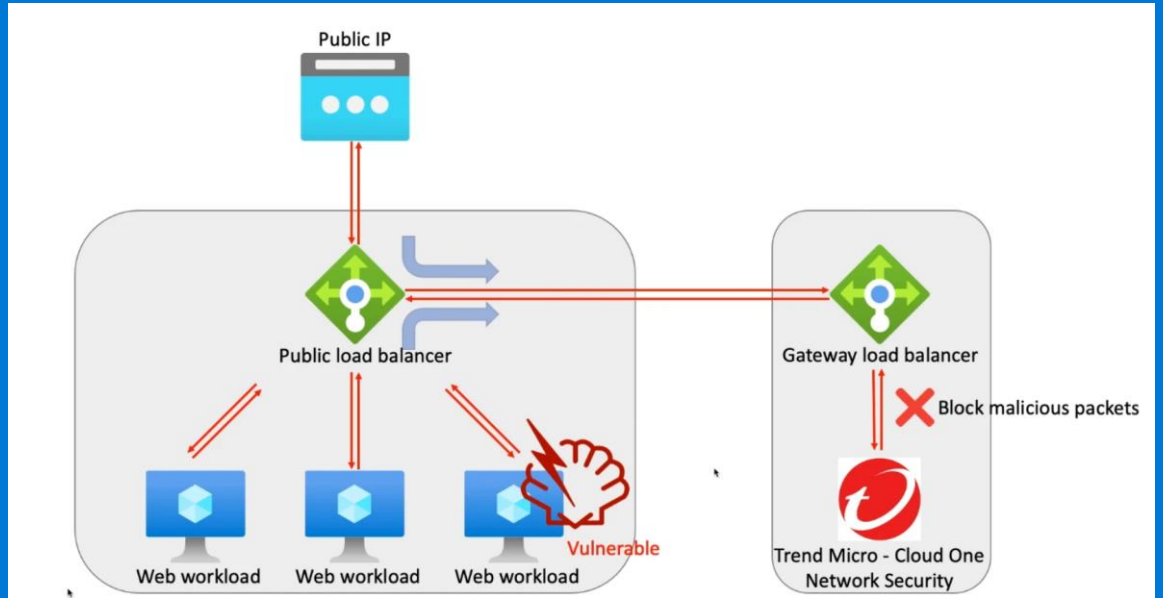
Example: **Trend Micro**

- Reduce complexity
- Simple deployment
- More customers

Before



After



Gateway Load Balancer partners

Virtual firewalls
Check Point
Cisco
F5
Fortinet
Palo Alto Networks

Traffic observability
cPacket Networks
Glasnostic

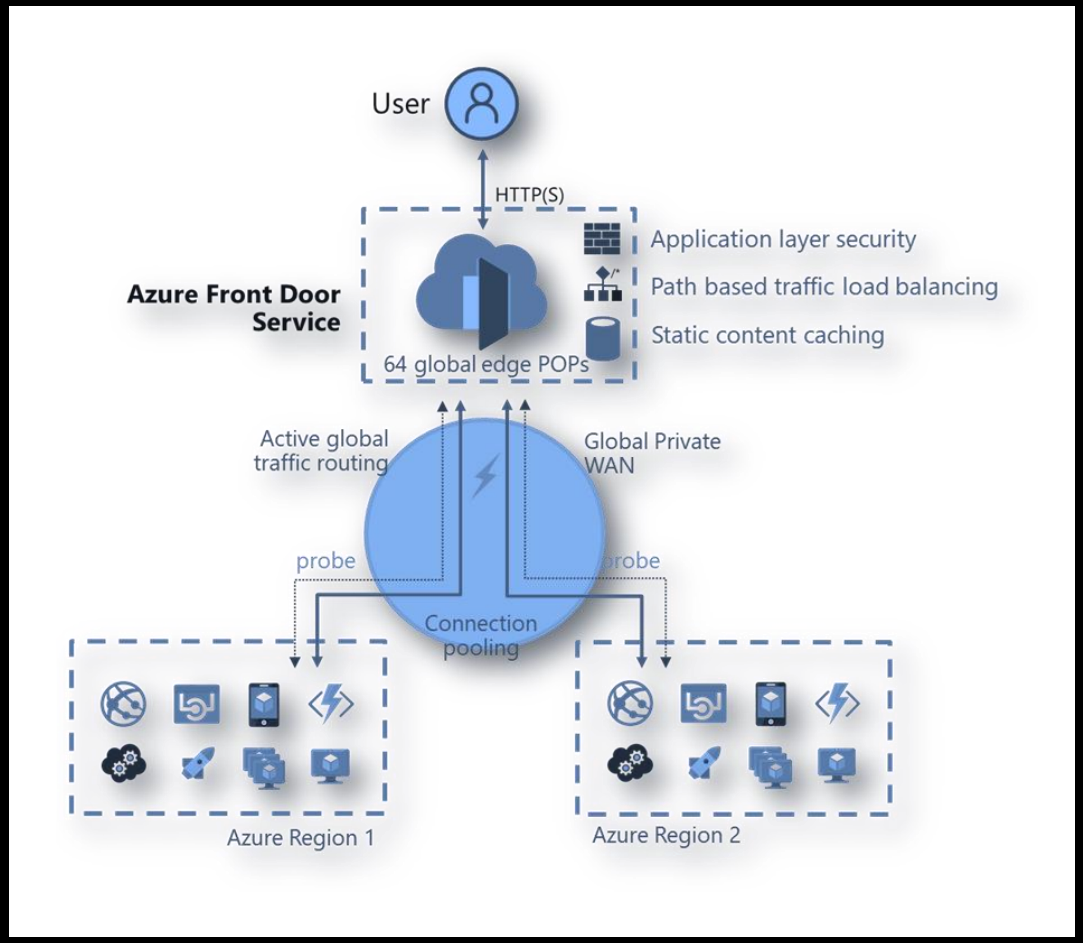
Network security
Citrix
Trend Micro
Valtix

DDoS protection
A10 Networks

Azure Front Door

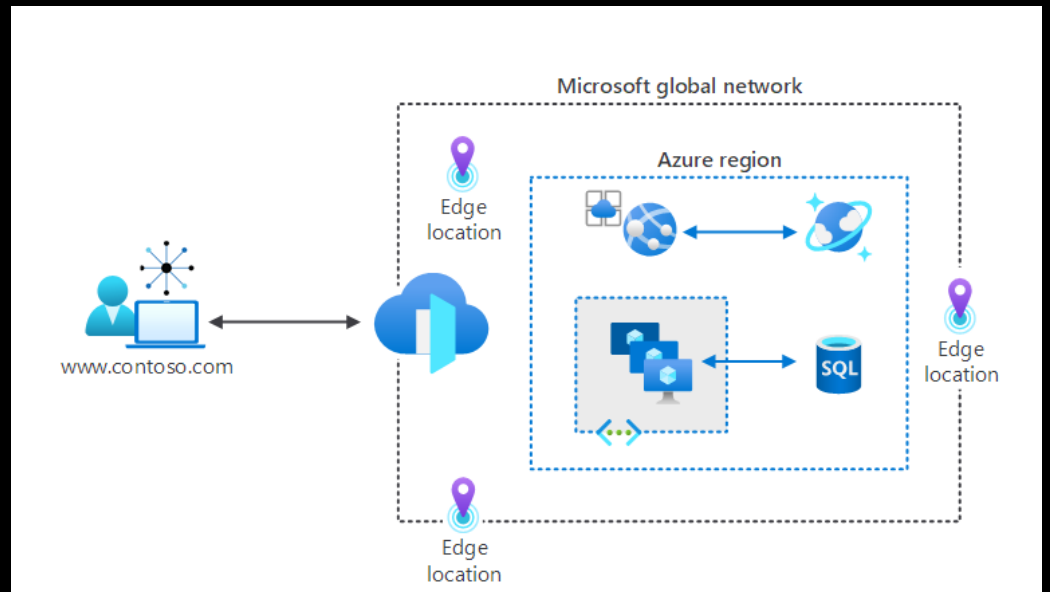
Azure Front Door

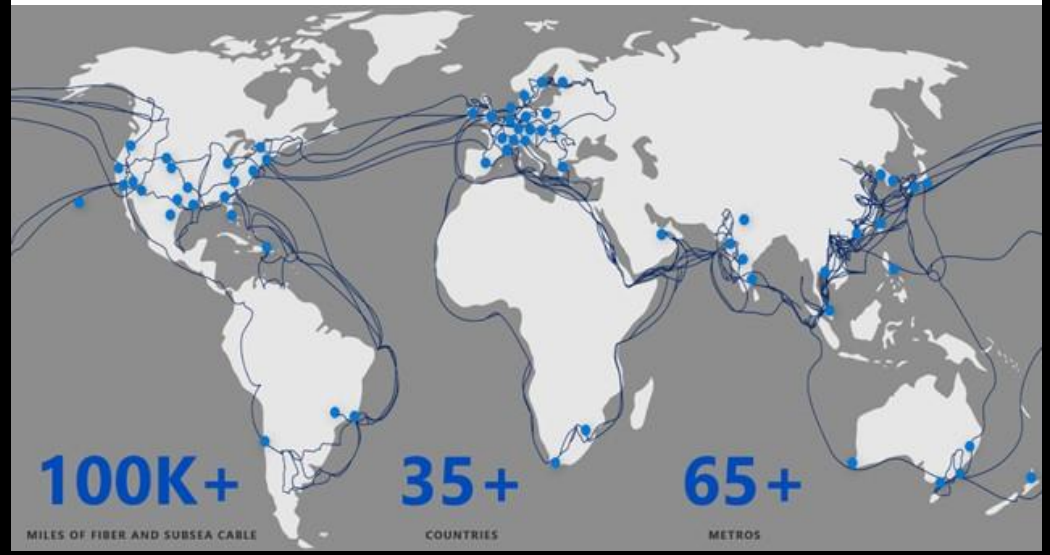
- Front Door works at Layer 7 (HTTP/HTTPS layer) using anycast protocol with split TCP and Microsoft's global network to improve global connectivity.
- Front Door provides a range of traffic-routing methods and backend health monitoring options to suit different application needs and automatic failover scenarios.
- Front Door is resilient to failures, including failures to an entire Azure region.
- SSL offload and application acceleration
- Web Application Firewall and DDoS protection
- CDN for caching
- Centralized traffic orchestration view



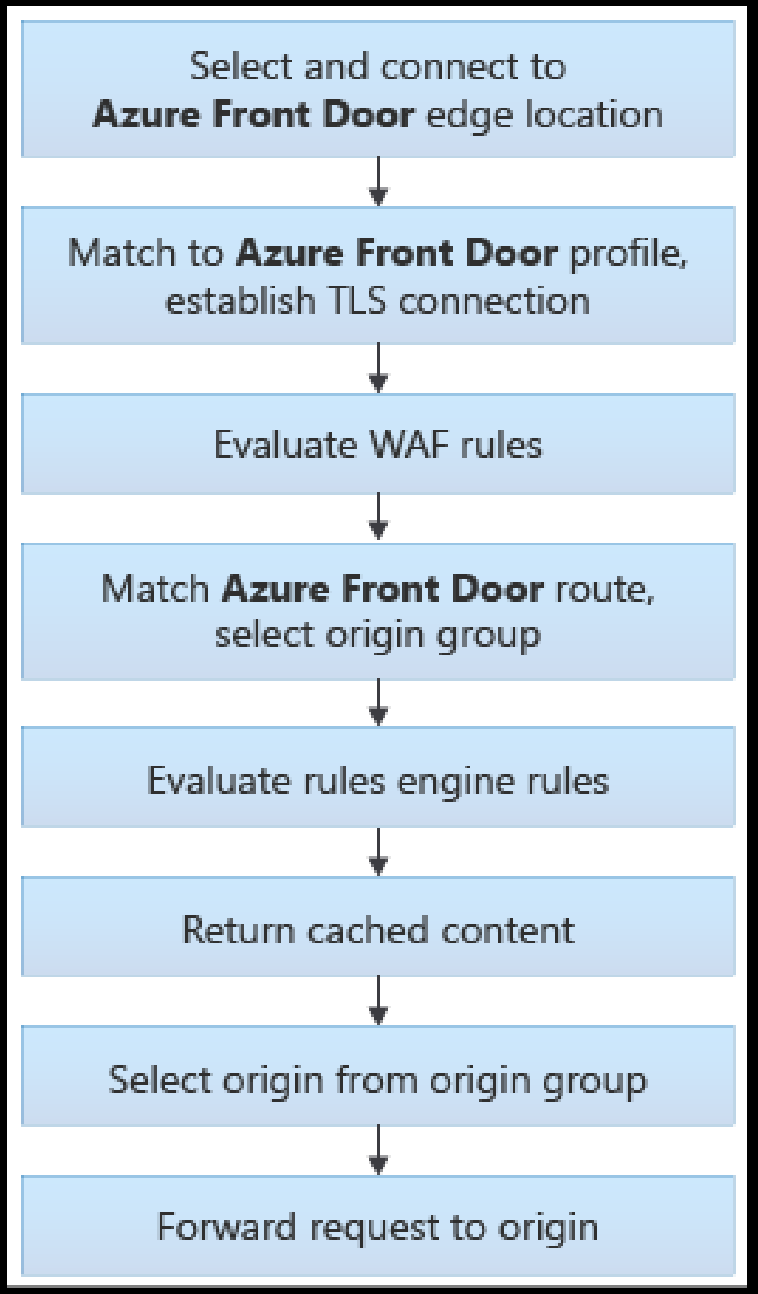
Azure Front Door - Features

- ✓ Accelerated application performance by using split TCP-based anycast protocol.
- ✓ Intelligent health probe monitoring for backend resources.
- ✓ URL-path based routing for requests.
- ✓ Enables hosting of multiple websites for efficient application infrastructure.
- ✓ Cookie-based session affinity.
- ✓ SSL offloading and certificate management.
- ✓ Define your own custom domain.
- ✓ Application security with integrated Web Application Firewall (WAF).
- ✓ Redirect HTTP traffic to HTTPS with URL redirect.
- ✓ Custom forwarding path with URL rewrite.
- ✓ Native support of end-to-end IPv6 connectivity and HTTP/2 protocol.





Azure Front Door – Routing Architecture Overview

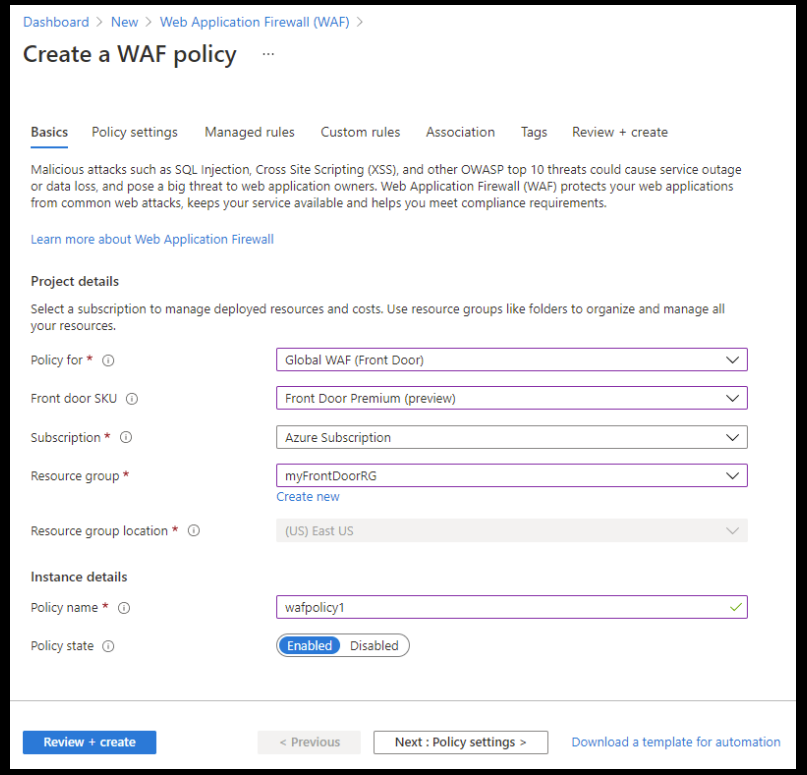


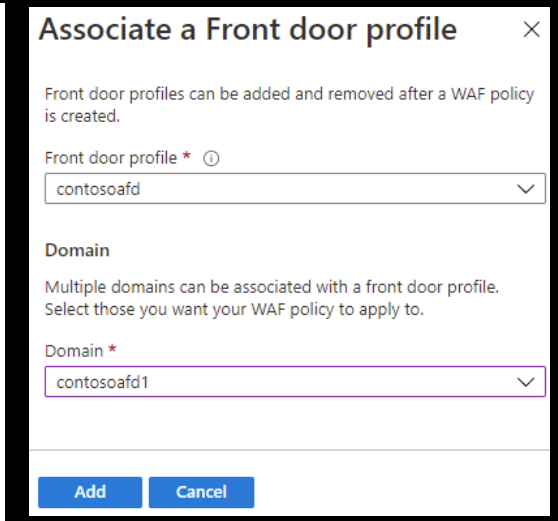
Azure Front Door - WAF

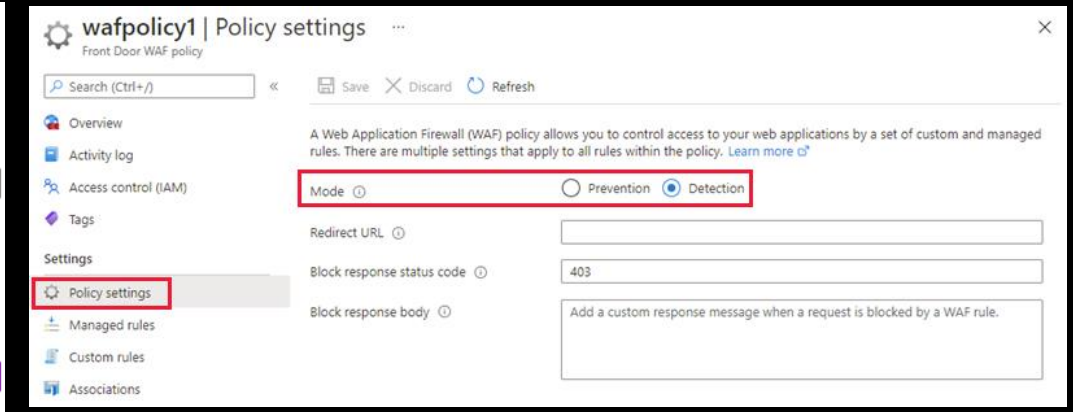
- **Policy settings –** Detection / Prevention
- **Managed rule sets**
- **Custom rules**
- **Exclusion Lists**
- **Geo-Filtering**
- **Bot Protection**
- **IP Restriction**
- **Rate Limiting**
- **Tuning**
- **Monitoring and Logging**

Azure Front Door - WAF

Create-Associate-Configure a Web Application Firewall policy on Azure Front Door







Azure Front Door – Route Configuration

A Front Door Standard/Premium routing configuration is composed of two major parts: "left-hand side" and "right-hand side"

Azure front Door match the incoming request to the left-hand side of the route and the right-hand side defines how it process the request

Frontend host matching

When matching Frontend hosts, we use the logic defined below:

- ✓ Look for any routing with an exact match on the host.
- ✓ If no exact frontend hosts match, reject the request and send a 400 Bad Request error.

Path matching

Use a similar logic as frontend hosts:

- ✓ Look for any routing rule with an exact match on the Path
- ✓ If no exact match Paths, look for routing rules with a wildcard Path that matches
- ✓ If no routing rules are found with a matching Path, then reject the request and return a 400: Bad Request error HTTP response.

Frontend host matching

Routing rule	Frontend hosts	Path
A	foo.contoso.com	/*
B	foo.contoso.com	/users/*
C	www.fabrikam.com, foo.adventure-works.com	/*, /images/*

If the following incoming requests were sent to Front Door, they would match against the following routing rules from above:

Incoming frontend host	Matched routing rule(s)
foo.contoso.com	A, B
www.fabrikam.com	C
images.fabrikam.com	Error 400: Bad Request
foo.adventure-works.com	C
contoso.com	Error 400: Bad Request
www.adventure-works.com	Error 400: Bad Request
www.northwindtraders.com	Error 400: Bad Request

Path matching

Routing rule	Frontend host	Path
A	www.contoso.com	/
B	www.contoso.com	/*
C	www.contoso.com	/ab
D	www.contoso.com	/abc
E	www.contoso.com	/abc/
F	www.contoso.com	/abc/*
G	www.contoso.com	/abc/def
H	www.contoso.com	/path/

Given that configuration, the following example matching table would result:

Incoming Request	Matched Route
www.contoso.com/	A
www.contoso.com/a	B
www.contoso.com/ab	C

Azure Front Door – Routing Methods

There are four traffic routing methods available in Front Door:

- Latency The latency-based routing ensures that requests are sent to the lowest latency backends acceptable within a sensitivity range. Basically, your user requests are sent to the "closest" set of backends in respect to network latency.

- Priority You can assign priorities to your backends when you want to configure a primary backend to service all traffic. The secondary backend can be a backup in case the primary backend becomes unavailable.

- Weighted You can assign weights to your backends when you want to distribute traffic across a set of backends evenly or according to the weight coefficients. Traffic is distributed as per weights if the latencies of the backends are within the acceptable latency sensitivity range in the backend pool.

- Session Affinity You can configure session affinity for your frontend hosts or domains to ensure requests from the same end user gets sent to the same backend.

Azure Front Door - Origins

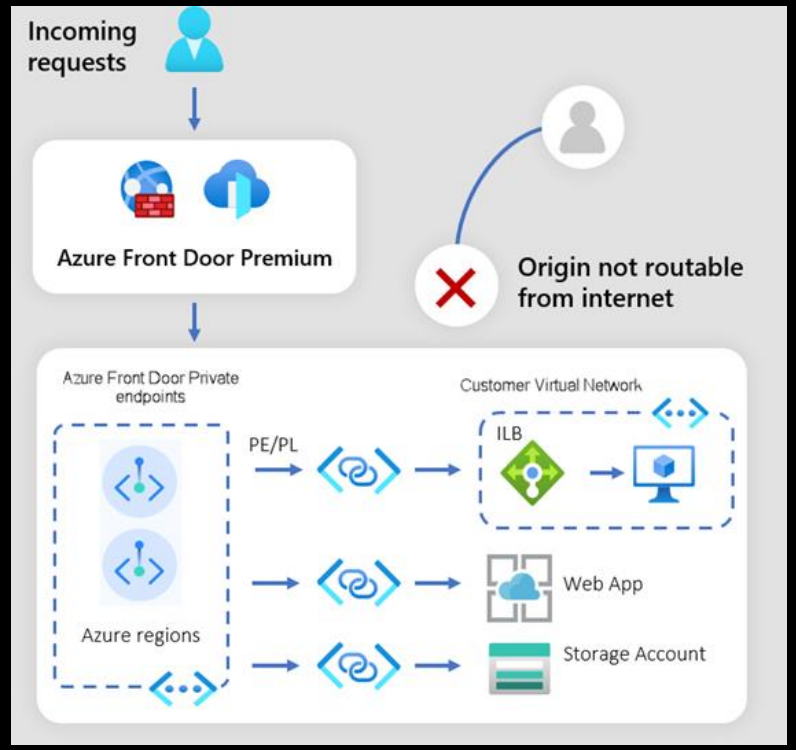
Azure Front Door Standard/Premium supports both Azure origins and also non-Azure origins, such as when your application is hosted in your on-premises datacenter or with another cloud provider.

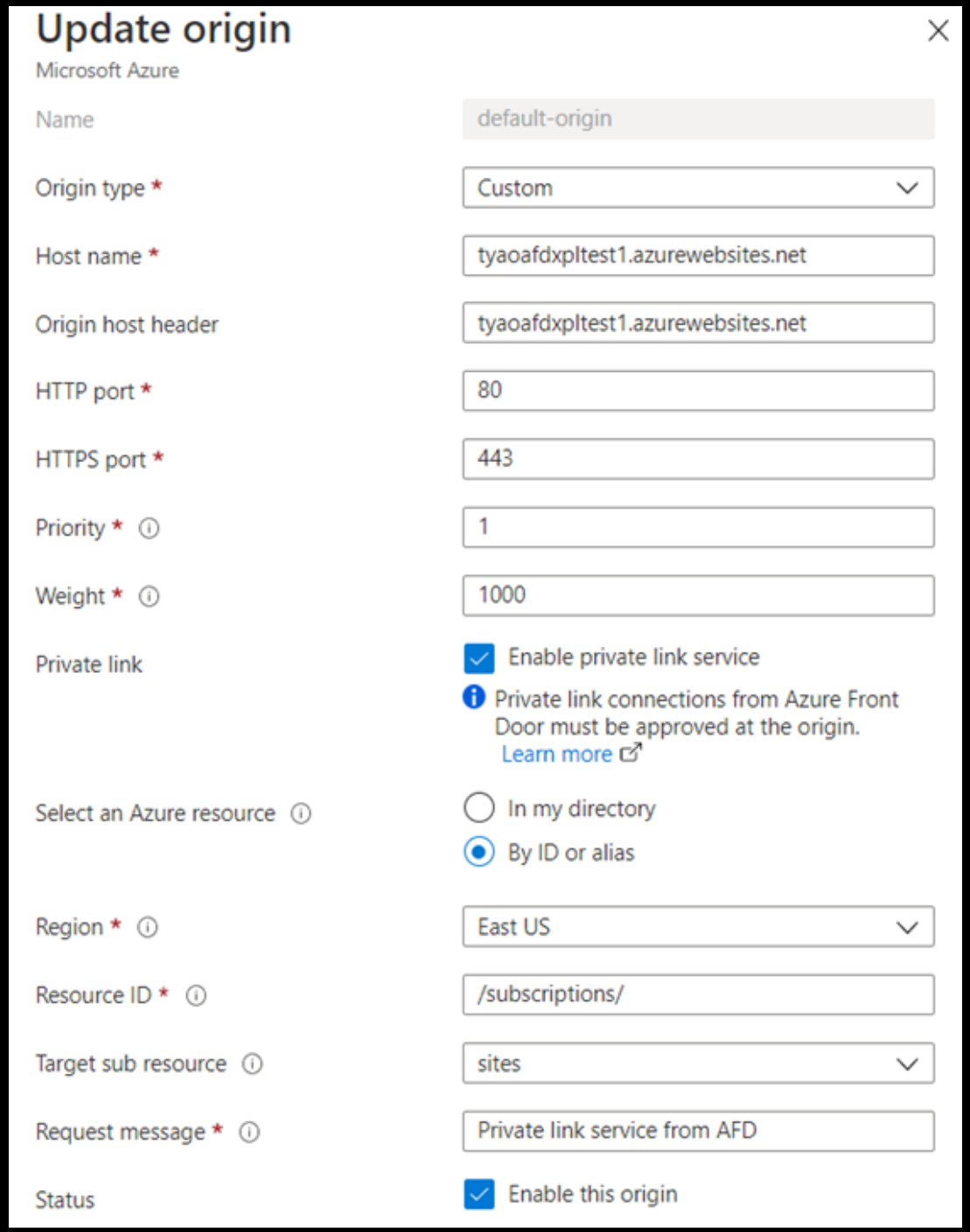
When you add an origin to an Azure Front Door Standard/Premium origin group, you must also add the following information:

- **Origin type**
- **Subscription and Origin host name**
- **Private Link**
- **Certificate Subject Name Validation**
- **Origin host header**
- **Priority**
- **Weight**

Azure Front Door – Private Link

- Azure Front Door Premium SKU can connect to your origin via private link service.
- Your applications can be hosted in your private VNet or behind a PaaS service such as Web App and Storage Account, removing the need for your origin to be publicly accessible.
- Origin support for direct private endpoint connectivity is currently limited to:
  - **Storage (Azure Blobs)**
  - **App Services**
  - **Internal load balancers**





Azure Front Door - DDoS

- ✓ **Integration with Azure DDoS Protection Basic**
  Default DDoS protection defends against the most common, frequently occurring layer 7, DNS query floods, and layer 3 and 4 volumetric attacks that target public endpoints.

- ✓ **Protocol blocking**
  Front Door only accepts traffic on the HTTP and HTTPS protocols and will only process valid requests with a known Host header.

- ✓ **Capacity absorption**
  Front Door is located at the edge of Azure's network, absorbing, and geographically isolating large volume attacks

- ✓ **Caching**
  Front Door's caching capabilities can be used to protect backends from large traffic volumes generated by an attack. Cached resources will be returned from the Front Door edge nodes so they don't get forwarded to your backend.

- ✓ **Web Application Firewall (WAF)**
  Front Door's Web Application Firewall (WAF) can be used to mitigate many different types of attacks.

Azure Front Door SKUs

Azure Front Door Standard SKU:

- Content delivery optimized
- Offering both static and dynamic content acceleration
- Global load balancing
- SSL offload
- Domain and certificate management
- Enhanced traffic analytics
- Basic security capabilities

Azure Front Door Premium SKU builds on capabilities of Standard SKU, and adds:

- Microsoft managed rule set
- BOT protection
- Private Link support
- Integration with Microsoft Threat Intelligence and security analytics.

Decision criteria:

Criteria	Analysis
Scalability	Does your organization scale out content? Organizations that host scalable content will benefit more from using Azure Front Door.
Pricing	Does your organization prefer a monthly charge for each policy or hourly billing? Do you want to pay extra charges for custom rules? Review the pricing considerations in the following section.
Content delivery	Do you require content optimization, without extensive security capabilities? Azure Front Door Standard is a good choice in this case.
Security	Do you have enhanced security requirements? Azure Front Door Premium is your best option.

Choose the Right Load Balancer -

Service	Global/regional	Recommended traffic
Azure Front Door	Global	HTTP(S)
Traffic Manager	Global	non-HTTP(S)
Application Gateway	Regional	HTTP(S)
Azure Load Balancer	Regional	non-HTTP(S)

Traffic Manager or Front Door?

Traffic Manager

Any protocol
Because Traffic Manager works at the DNS layer, you can route any type of network traffic; HTTP, TCP, UDP, etc.

On-premise routing
With routing at a DNS layer, traffic always goes from point to point. Routing from your branch office to your on-premises datacenter can take a direct path; even on your own network using Traffic Manager.

Billing format
DNS-based billing scales with your users and for services with more users, plateaus to reduce cost at higher usage.

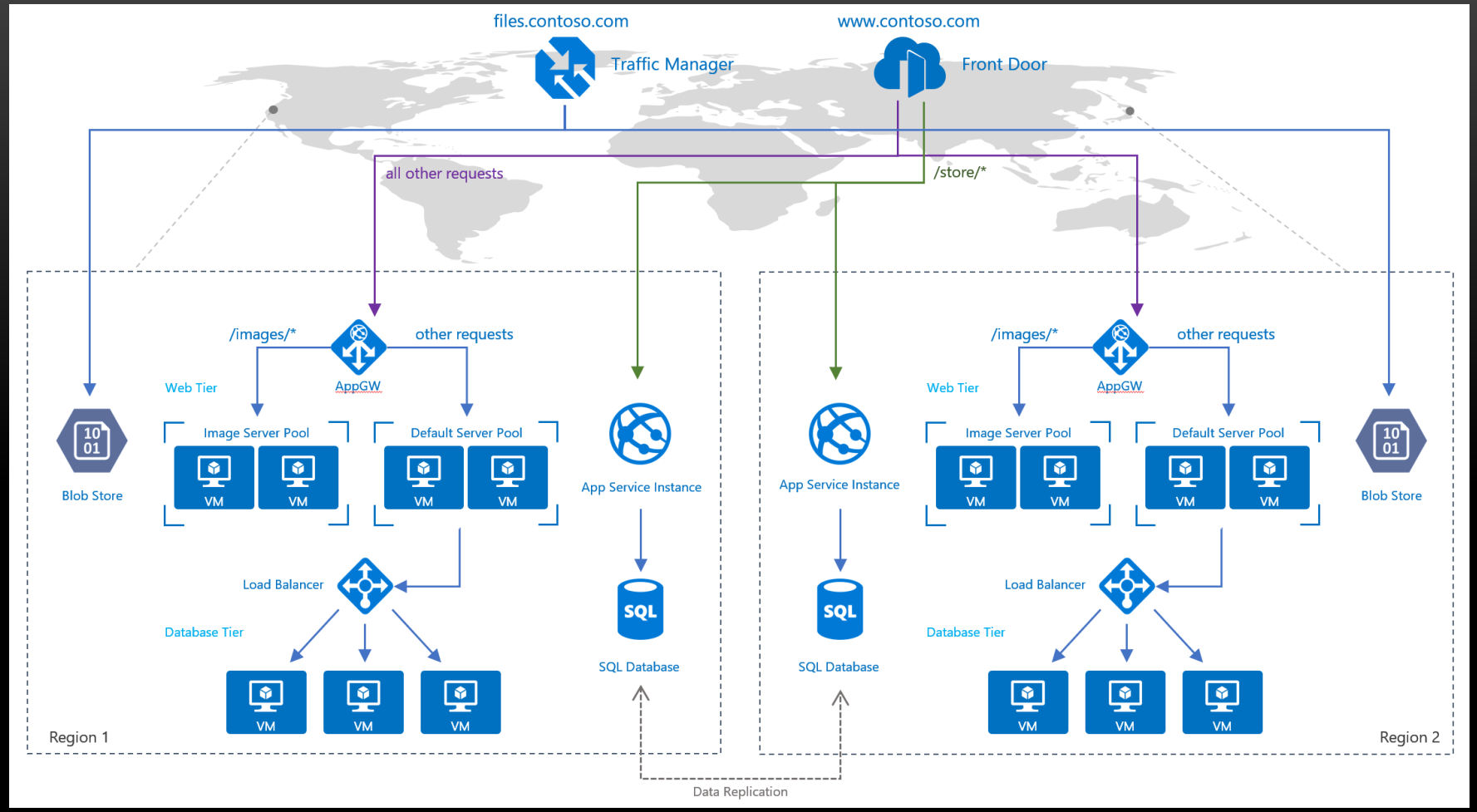
Front Door

HTTP acceleration
With Front Door traffic is proxied at the Edge of Microsoft's network. Because of this, HTTP(S) requests see latency and throughput improvements reducing latency for SSL negotiation and using hot connections from AFD to your application.

Independent scalability
Because Front Door works with the HTTP request, requests to different URL paths can be routed to different backend/regional service pools (microservices) based on rules and the health of each application microservice.

Inline security
Front Door enables rules such as rate limiting and IP ACL-ing to let you protect your backends before traffic reaches your application.

Traffic Manager or Front Door?




Microsoft Education