The bridge to possible

Traffic Inspection in Azure using Cisco Secure Firewall and Gateway Load Balancer

Sameer Singh
Technical Marketing Engineering
BRKSEC-2109

Welcome to the Multi-Cloud Era

Hashicorp 2022 State of Cloud Strategy Survey

5 Numbers To Remember

90%

Say multi-cloud is working

86%

Rely on cloud platform teams

94%

Are wasting money in the cloud

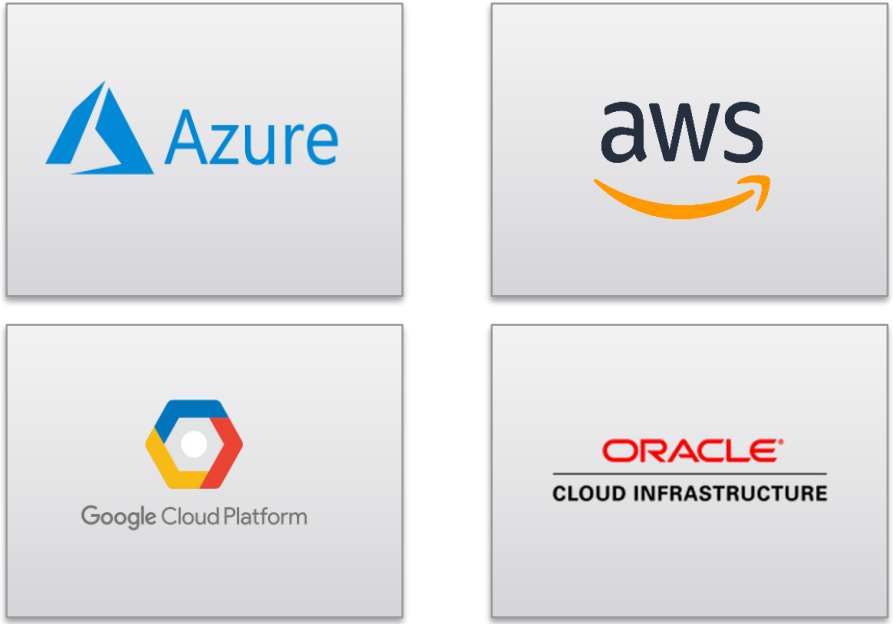
89%

See security as a key driver of cloud success

#1

Rank of skills shortages as a multi-cloud barrier

Major Cloud Providers



Abstract

In this session, we will see how the introduction of Gateway load balancer in Azure **simplifies the insertion** of Cisco Secure firewall in the Azure environment. We will look at the different components of the solution, how it can leverage autoscaling and addresses some of the current challenges.

Agenda

- Azure Load Balancers
- Load Balancer Challenges
- Gateway Load Balancer
- Configuration Overview
- Demo
- Automation and Auto scale Solution Overview
- Key Takeaway

About the Speaker

Sameer Pratap Singh

- B-Tech in Electronics and Communication Engineering
- Security Solutions Consulting Engineer till 2021
- Technical Marketing Engineer – Network Security
- Interested in everything automation
- CCIE Security

Cisco Webex App

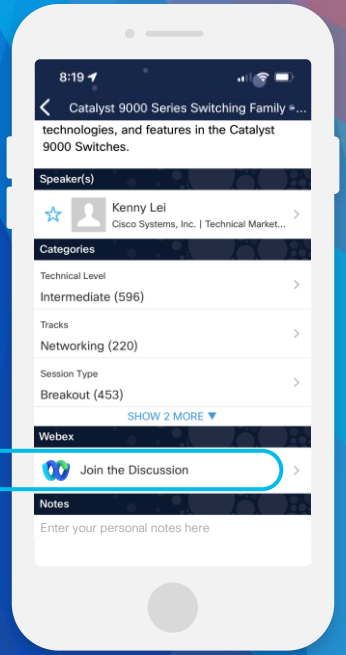
Questions?

Use Cisco Webex App to chat
with the speaker after the session

How

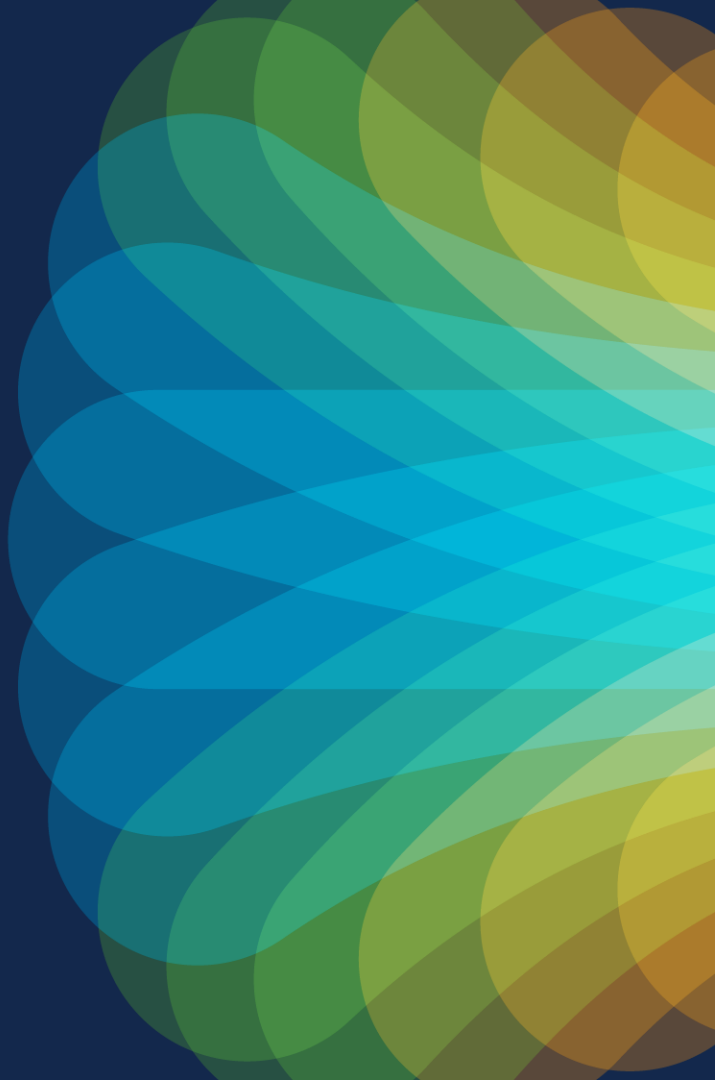
1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install the Webex App or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until June 9, 2023.



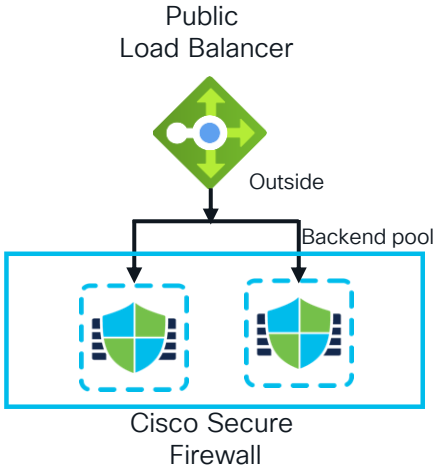
https://ciscolive.ciscoevents.com/ciscolivebot/#BRKSEC-2109

Azure Load Balancers Overview



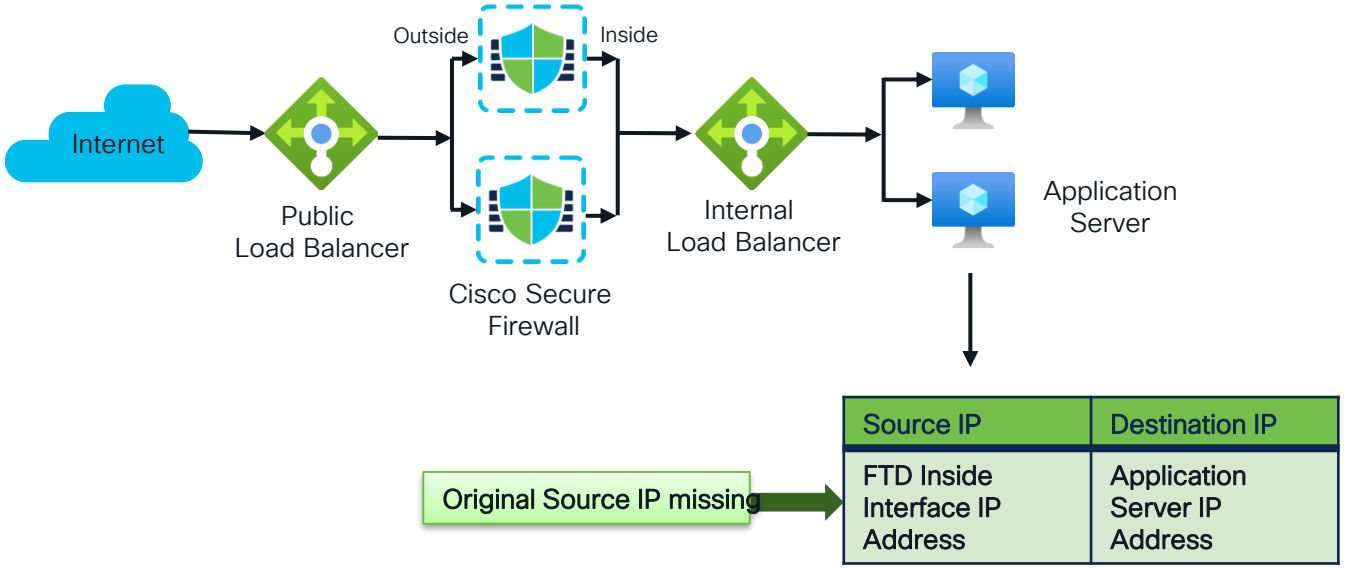
Azure – Standard Load Balancers

- Acts as a single point of contact and distributes incoming traffic across multiple instances
- Load Balancing rules decide traffic flow
- Improves scalability and availability of applications
- Health probes periodically check the health of the backend instances
- Types – Public and Internal Load Balancers

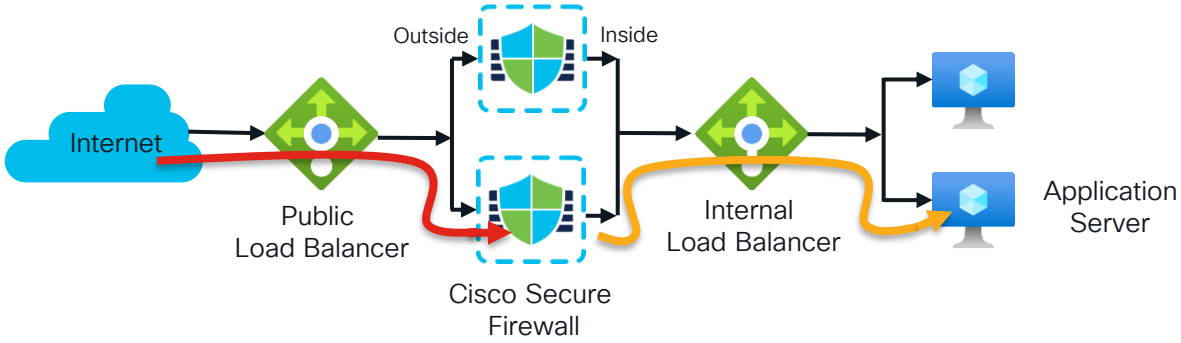


Standard Load Balancer Deployment
with Cisco Secure Firewall

- The original Initiator IP Address is unknown to the target application
- NAT and Route need to be configured on the firewall



Standard Load Balancer Deployment
with Cisco Secure Firewall



Source IP	Destination IP
Client IP	Public Load Balancer Frontend IP

Source IP	Destination IP
Client IP	Cisco Secure Firewall Outside Interface IP

Source IP	Destination IP
Cisco Secure Firewall Inside Interface IP	Internal Load Balancer frontend IP

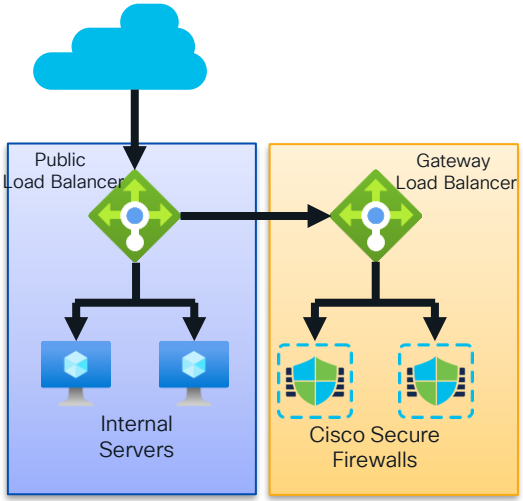
Source IP	Destination IP
Cisco Secure Firewall Inside Interface IP	Application Server IP

Standard Load Balancer Challenges

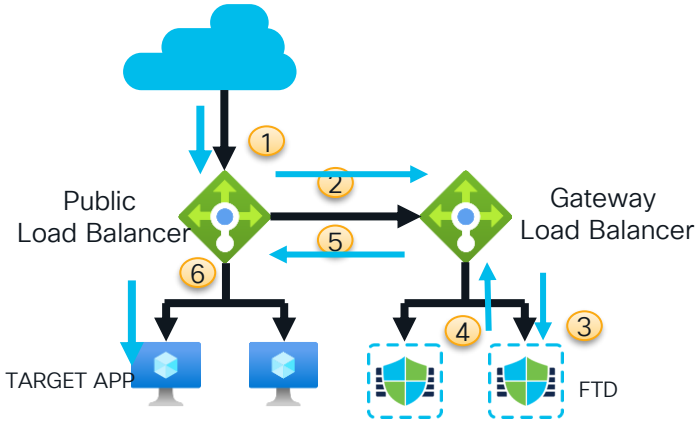
- Management Overhead
- The Source IP address of the packet is hidden
- Might Require to rearchitect the environment
- Operational Complexity

Azure – Gateway Load Balancer

- A load balancer solution which simplifies insertion of network firewall service in Azure environment.
- Acts like a **Bump-in-the-wire**.
- Redirection with VXLAN protocol
- Firewall receives and forwards traffic through the same interface



Traffic Flow



Source IP: Original Initiator IP Address
Destination IP: Application Server IP Address

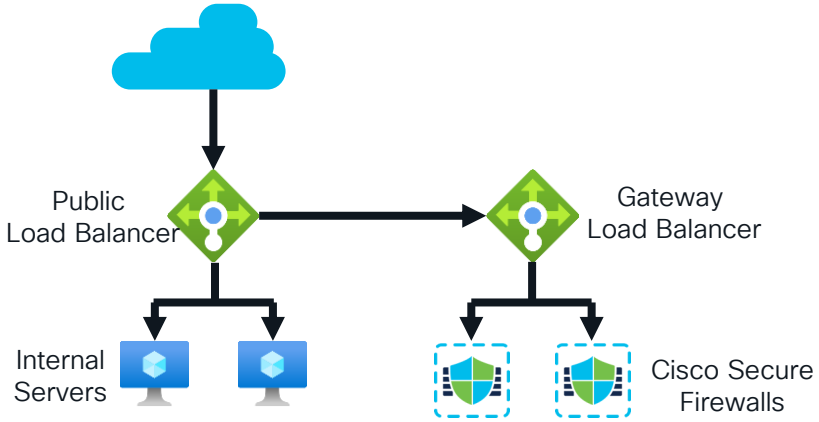
- NAT and Route are not required to be configured on the firewall
- GWLB maintains flow stickiness to a specific instance in the backend pool

1. Inbound traffic reached the Public IP of the load balancer
2. Load balancer forwards the traffic to the Gateway Load balancer
3. GWLB forwards the traffic to one of the firewall instances in the backend pool for inspection
4. Firewall returns inspected traffic to GWLB
5. GWLB returns traffic to the load balancer
6. Load balancer forwards it to the internal server

Service Chaining

- GWLB can be Chained to

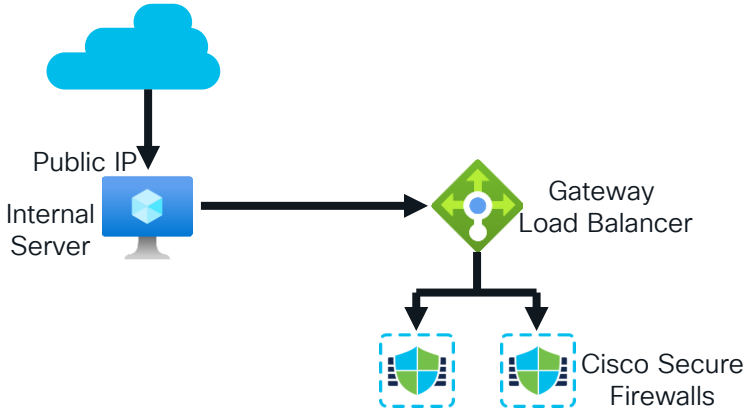
A Standard Public Load Balancer



Service Chaining

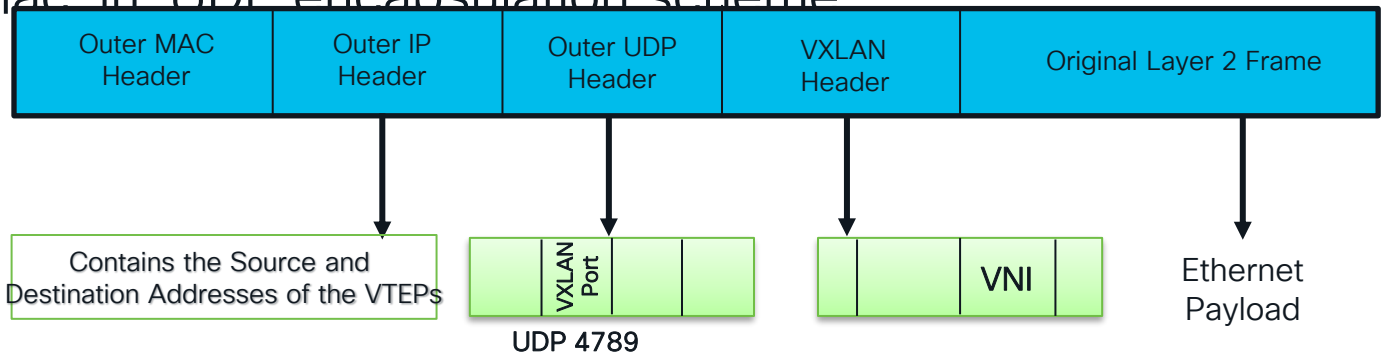
- GWLB can be Chained to

A Standard Public IP attached to a Virtual Machine



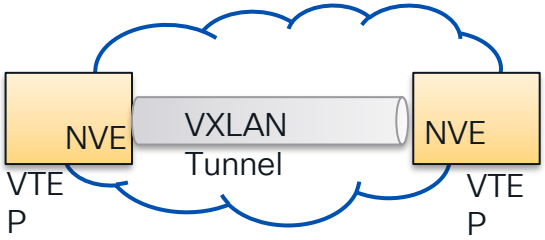
Virtual Extensible LAN (VXLAN) – Overview

- Provide VLAN functionality with greater extensibility and flexibility
- Extends layer 2 segments over the underlying layer 3 network infrastructure
- The transport protocol used is IP plus UDP
- Mac-in-UDP encapsulation scheme



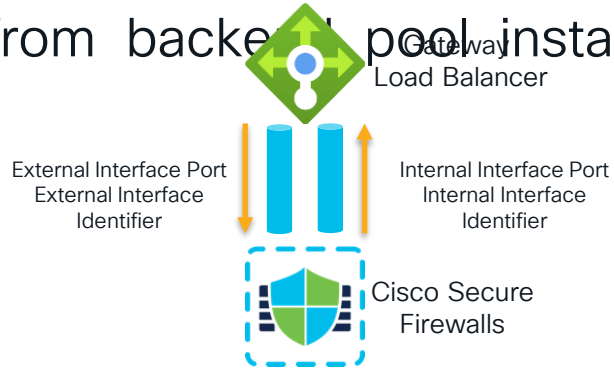
VXLAN Components

- Network Virtualization Edge (NVE) - Logical interface where the encapsulation and de-encapsulation occur
- VXLAN Tunnel Endpoint (VTEP) - This is the device that does the encapsulation and de-encapsulation
- VXLAN Network Identifier (VNI) – 24-bit segment ID that defines the broadcast domain. Interchangeable with "VXLAN Segment ID"

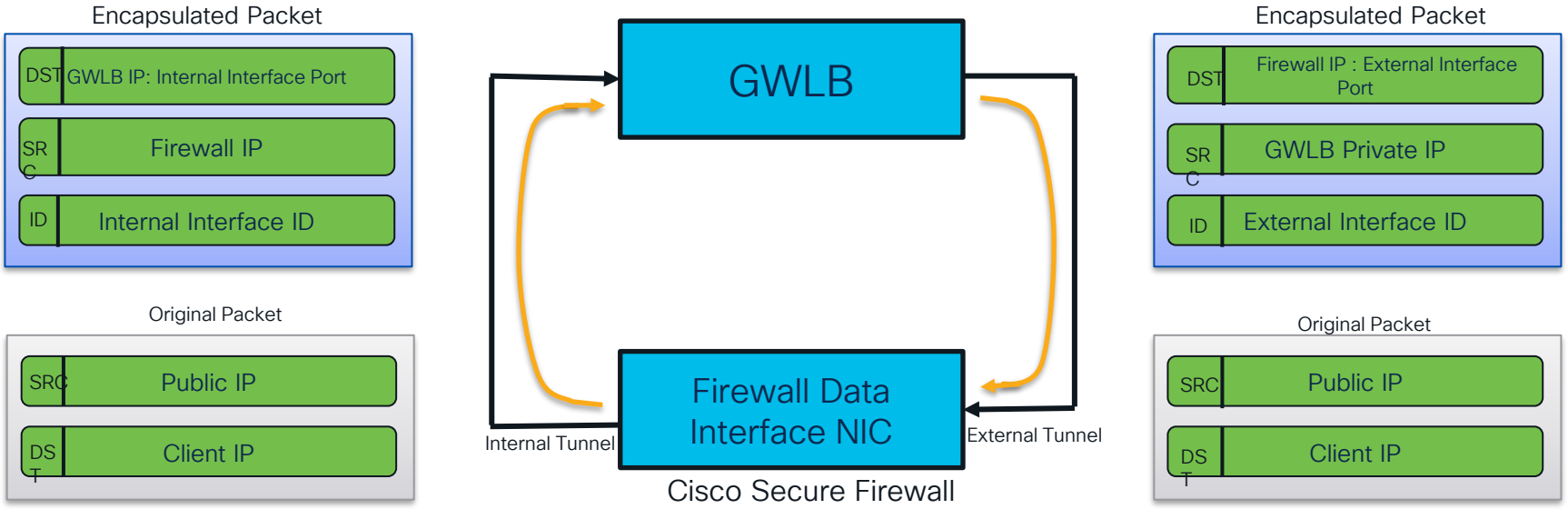


GWLB VXLAN Tunnels

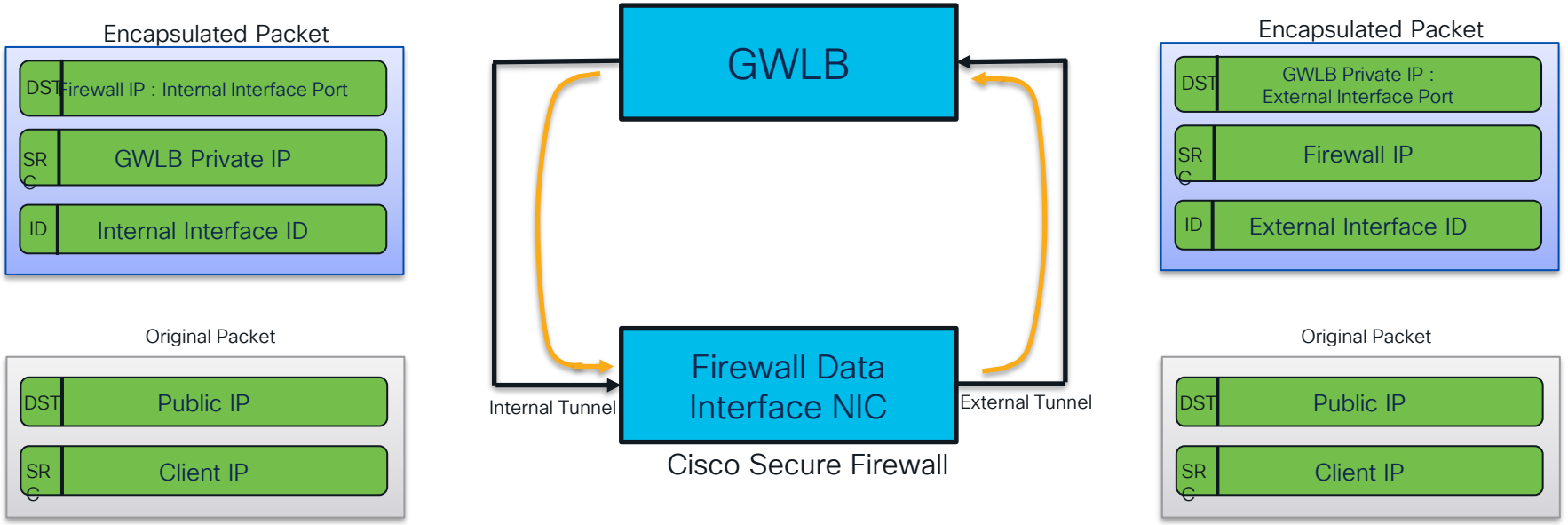
- Azure GWLB uses two VXLAN tunnels to communicate with its backend pool
- External Tunnel for untrusted traffic from GWLB to backend pool instance
- Internal Tunnel for trusted traffic from backend pool instance to GWLB
- Each Tunnel uses a different UDP Port
- Each Tunnel uses a different VNI



Traffic to the Client



Traffic from the Client



Encapsulated packets on the VTEP interface

```
   1: 07:00:30.275086       10.19.2.5.50447 > 10.19.2.6.10801:  udp 62
   2: 07:00:30.275239       10.19.2.6.56840 > 10.19.2.5.10800:  udp 62
   3: 07:00:30.276352       10.19.2.5.64198 > 10.19.2.6.10800:  udp 74
   4: 07:00:30.276429       10.19.2.6.50764 > 10.19.2.5.10801:  udp 74
   5: 07:00:32.286804       10.19.2.5.43481 > 10.19.2.6.10801:  udp 62
```

Original packets on the VNI interface

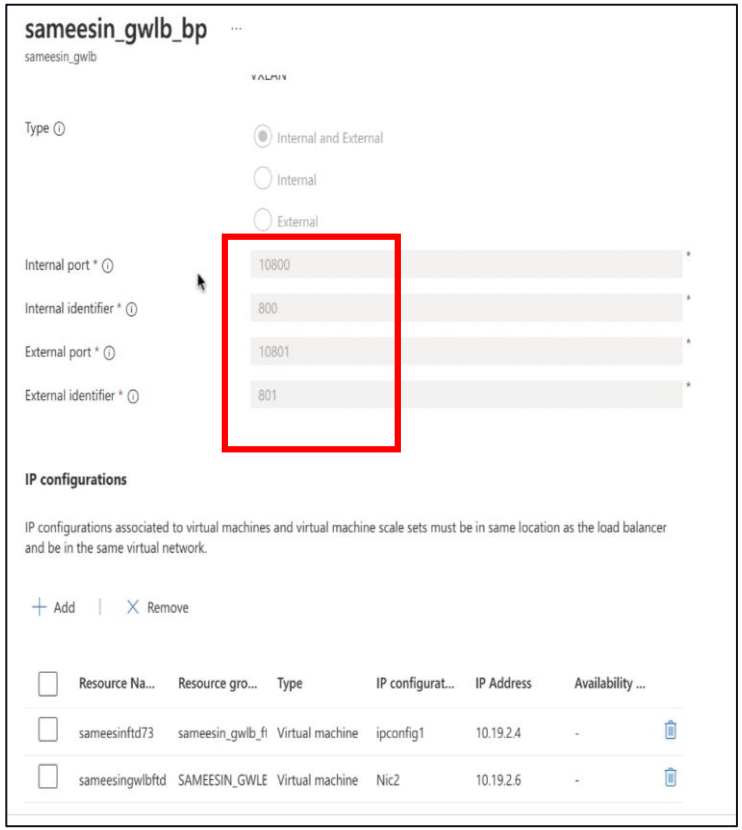
```
   1: 07:04:04.207920       49.37.41.50.50479 > 20.157.64.169.80: S 1579459360:1579459360(0) win 65535 <mss 1460,nop,wscale 6,nop,nop,timestamp 2391876
71 0,sackOK,eol>
   2: 07:04:04.208225       49.37.41.50.50479 > 20.157.64.169.80: S 2274994639:2274994639(0) win 65535 <mss 1380,nop,wscale 6,nop,nop,timestamp 2391876
71 0,sackOK,eol>
   3: 07:04:04.210651       20.157.64.169.80 > 49.37.41.50.50479: S 2880872722:2880872722(0) ack 2274994640 win 28960 <mss 1420,sackOK,timestamp 520348
 239187671,nop,wscale 9>
   4: 07:04:04.210758       20.157.64.169.80 > 49.37.41.50.50479: S 351140814:351140814(0) ack 1579459361 win 28960 <mss 1380,sackOK,timestamp 520348 2
39187671,nop,wscale 9>
   5: 07:04:04.439644       49.37.41.50.50479 > 20.157.64.169.80: . ack 351140815 win 2052 <nop,nop,timestamp 239187903 520348>
   6: 07:04:04.439705       49.37.41.50.50479 > 20.157.64.169.80: P 1579459361:1579459914(553) ack 351140815 win 2052 <nop,nop,timestamp 239187903 5203
48>
   7: 07:04:04.439827       49.37.41.50.50479 > 20.157.64.169.80: . ack 2880872723 win 2052 <nop,nop,timestamp 239187903 520348>
   8: 07:04:04.440055       49.37.41.50.50479 > 20.157.64.169.80: P 2274994640:2274995193(553) ack 2880872723 win 2052 <nop,nop,timestamp 239187903 520
348>
```

Azure GWLB Components

- **Frontend IP Configuration** – A private IP Address assigned to the Gateway Load balancer
- **Backend Pool** – Group of virtual machines that receive the incoming traffic from the Gateway load balancer
- **Load balancing rules (HA Port rule)** – Enables load balancing on all ports for TCP and UDP protocols.
- **Health Probe** – Used to identify healthy virtual machines in the backend pool to receive load-balanced traffic

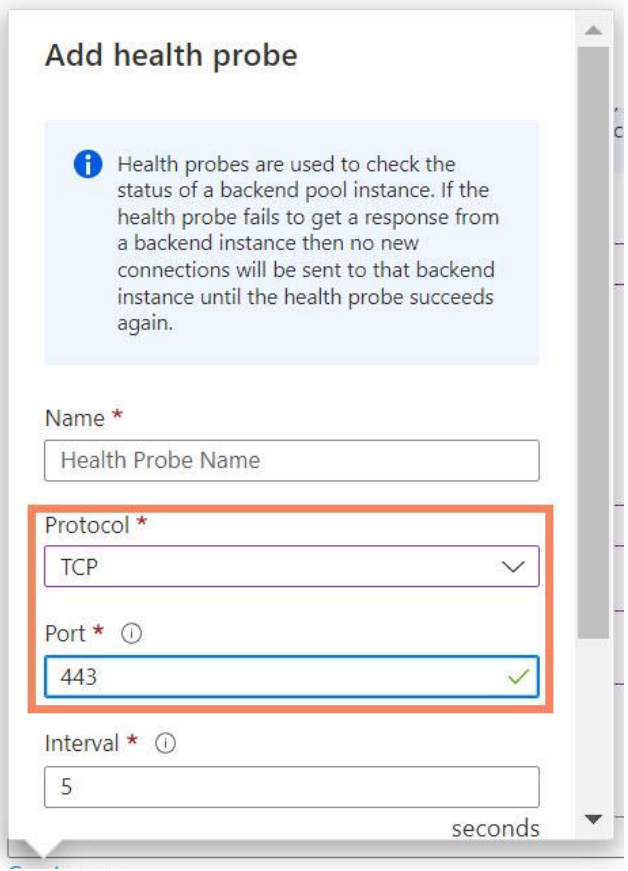
Backend Pool

- Defines the group of firewall instances that will inspect traffic for a given load-balancing rule
- Associate the VM NIC that should be part of the backend pool.
- Two VXLAN tunnels are defined for the backend pool
  - Internal Port and Internal Identifier
  - External Port and External Identifier



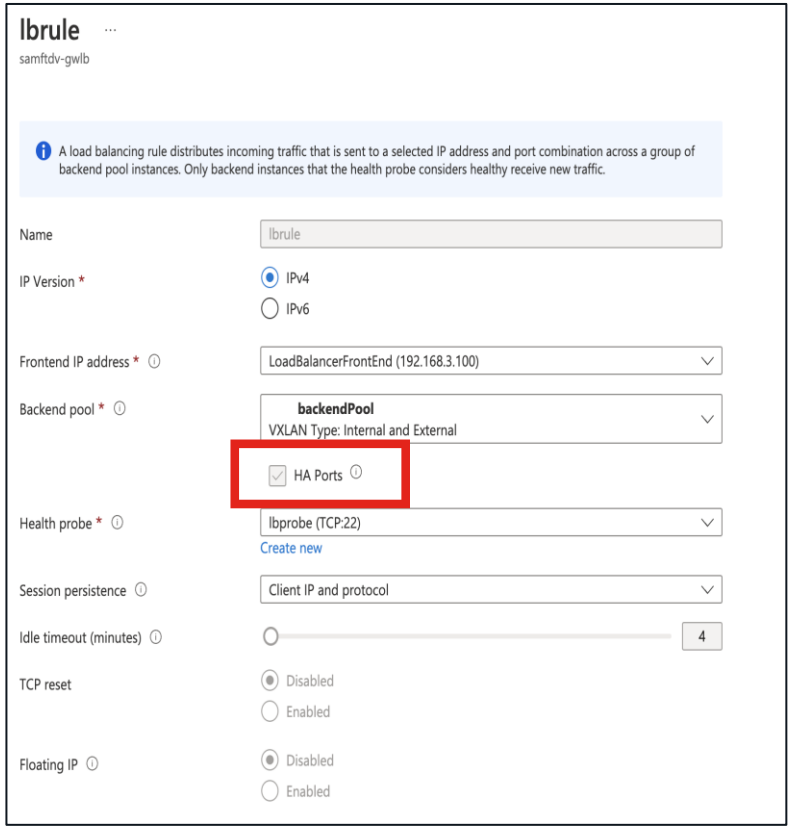
Health Probe

- Determines which backend pool instance will not receive new connections
- Defines the port and protocol to be used for the probe
- For Cisco Secure firewall, ports TCP/22 or TCP/443 can be used
- Defines the interval between probes

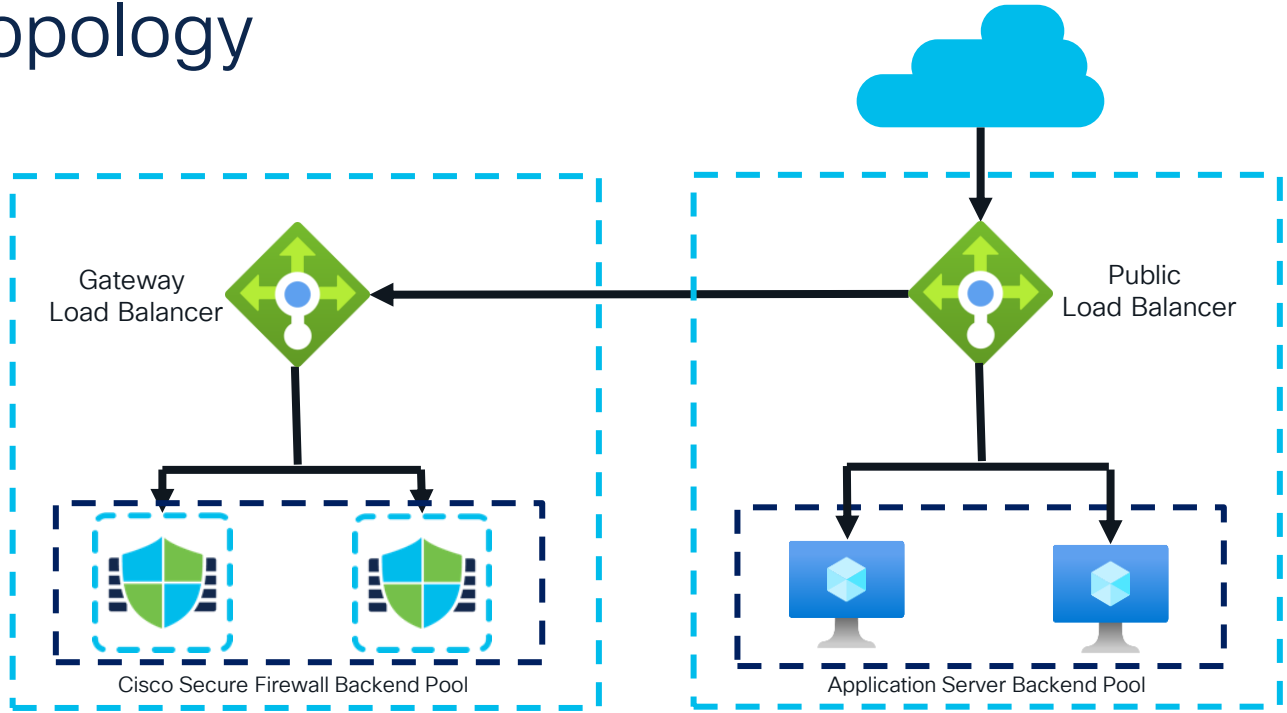


Load Balancer Rule

- Defines forwarding of traffic from the load balancer to instances in the backend pool
- GWLB allows only High Availability Ports rule
  - All forwarded traffic to the load balancer will match this rule
  - protocol - all and port – 0



Topology



Prerequisites

- Public Load Balancers (PLB) frontend IP configurations must be standard SKU.
- The network interface must have a standard SKU public IP address associated to it.

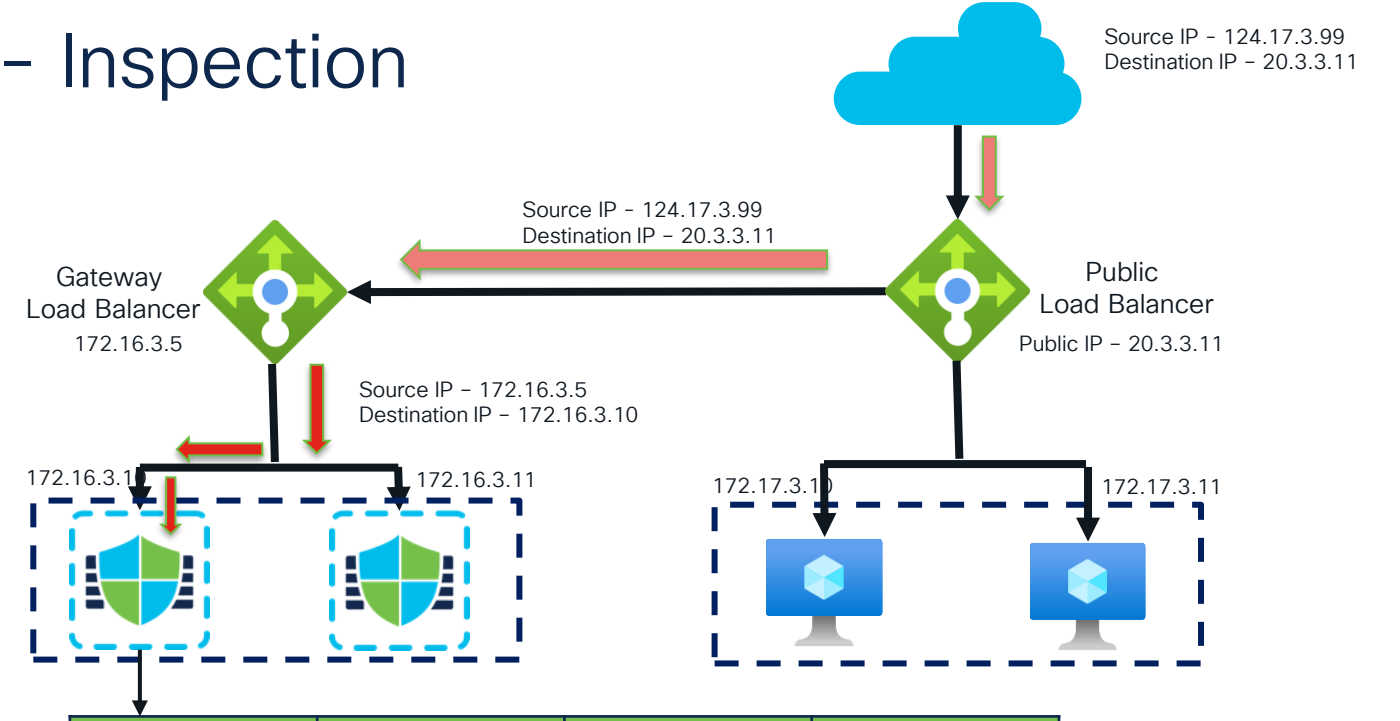
Enable service chaining by referencing the Gateway Load balancer to the Load balancer frontend IP or the public IP of the virtual machine

Cisco Secure Firewall Configuration

Prerequisites

- FMC and FTD versions should be 7.3 or above
- Secure Firewalls to be part of the backend pool should be registered to the Secure Firewall Management Center
- Only one data interface is required for this setup

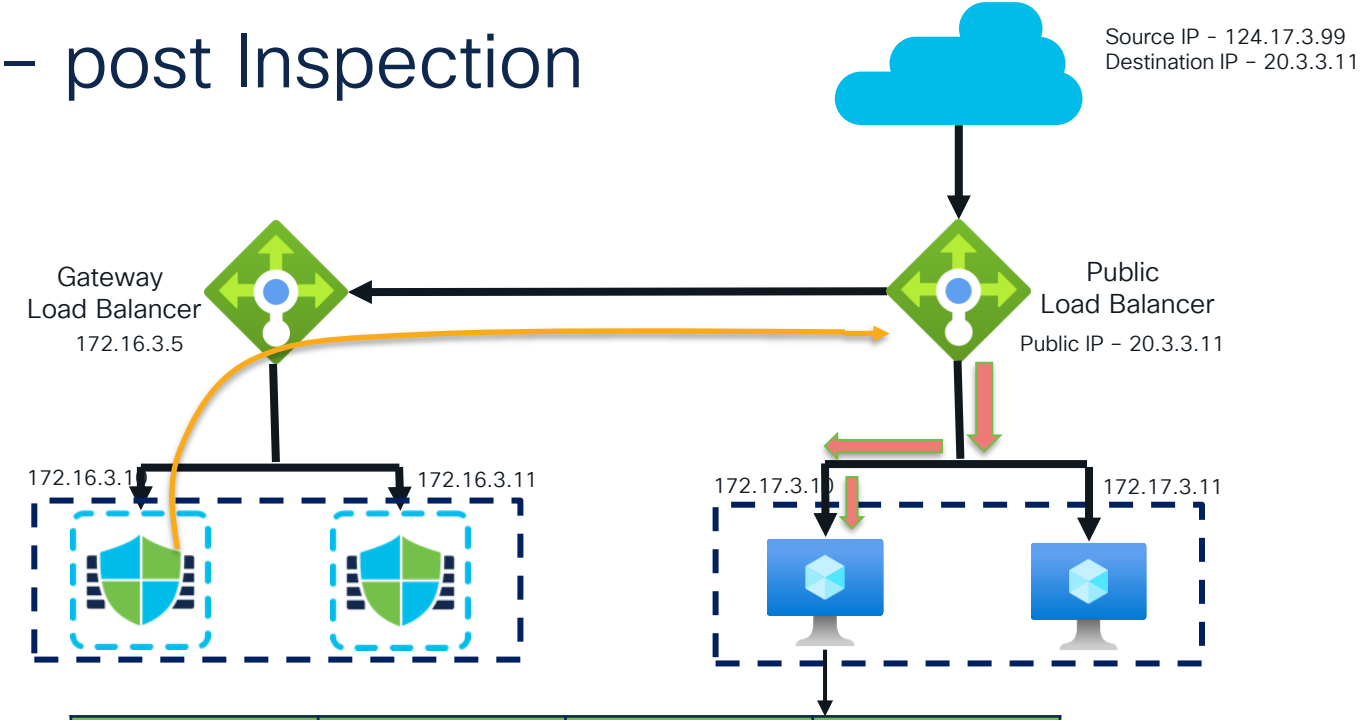
Flow – Inspection



Interface	Source IP	Destination IP	Destination Port
outside	172.16.3.5	172.16.3.10	10801
vni	124.17.3.99	20.3.3.11	80

Original Packet
Encapsulated Packet

Flow – post Inspection

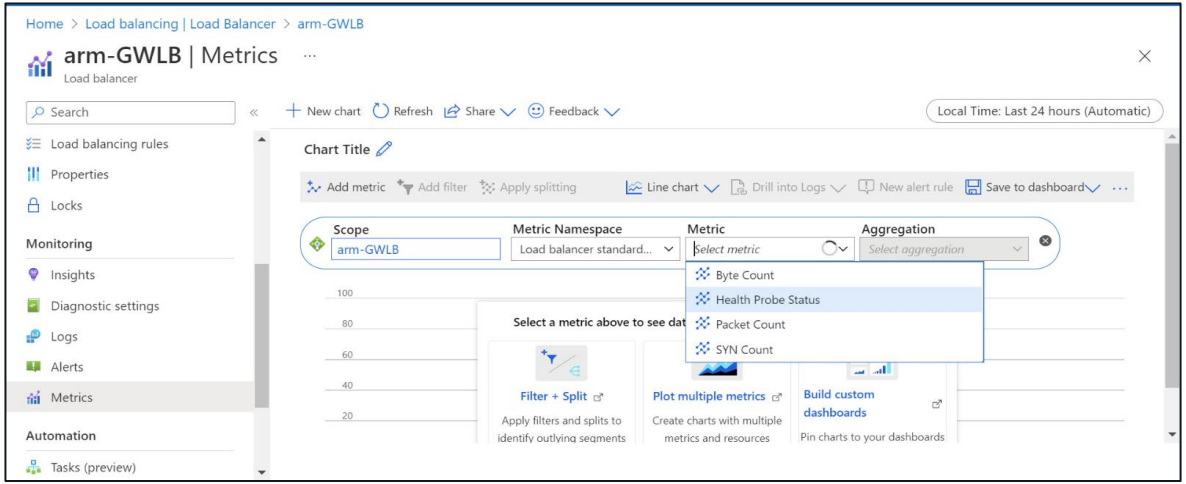


Interface	Source IP	Destination IP	Destination Port
eth0	124.17.3.99	172.17.3.10	80

Demo

Troubleshooting Tips

- Check the health probe status of the firewalls
- Navigate to **Metrics** in the GWLB section and select **Health Probe Status** as the metric



Troubleshooting Tips

- Packet capture on the FTDs
- Capture traffic on the VTEP interface to verify encapsulated traffic and health probe are being received.

```
1: 07:00:30.275086       10.19.2.5.50447 > 10.19.2.6.10801:  udp 62
2: 07:00:30.275239       10.19.2.6.56840 > 10.19.2.5.10800:  udp 62
3: 07:00:30.276352       10.19.2.5.64198 > 10.19.2.6.10800:  udp 74
4: 07:00:30.276429       10.19.2.6.50764 > 10.19.2.5.10801:  udp 74
5: 07:00:32.286804       10.19.2.5.43481 > 10.19.2.6.10801:  udp 62
```

Troubleshooting Tips

- If you see no traffic, check
  - GWLB configuration
  - Inbound effective security rules on the VTEP network interface
  - Interface configuration on the firewall
  - Confirm that the GWLB is associated with the firewall
- If you see no response for the health probe
  - check platform settings on the firewall

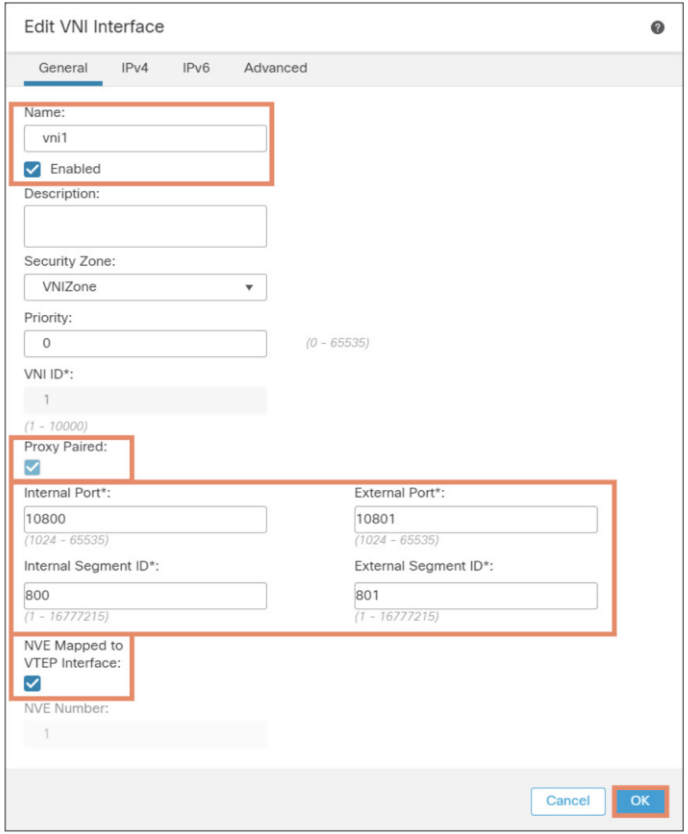
Troubleshooting Tips

- Packet capture on the FTDs
- Capture traffic on the VNI interface to verify traffic is received by the firewall.

```
   1: 07:04:04.207920       49.37.41.50.50479 > 20.157.64.169.80: S 1579459360:1579459360(0) win 65535 <mss 1460,nop,wscale 6,nop,nop,timestamp 2391876
71 0,sackOK,eol>
   2: 07:04:04.208225       49.37.41.50.50479 > 20.157.64.169.80: S 2274994639:2274994639(0) win 65535 <mss 1380,nop,wscale 6,nop,nop,timestamp 2391876
71 0,sackOK,eol>
   3: 07:04:04.210651       20.157.64.169.80 > 49.37.41.50.50479: S 2880872722:2880872722(0) ack 2274994640 win 28960 <mss 1420,sackOK,timestamp 520348
 239187671,nop,wscale 9>
   4: 07:04:04.210758       20.157.64.169.80 > 49.37.41.50.50479: S 351140814:351140814(0) ack 1579459361 win 28960 <mss 1380,sackOK,timestamp 520348 2
39187671,nop,wscale 9>
   5: 07:04:04.439644       49.37.41.50.50479 > 20.157.64.169.80: . ack 351140815 win 2052 <nop,nop,timestamp 239187903 520348>
   6: 07:04:04.439705       49.37.41.50.50479 > 20.157.64.169.80: P 1579459361:1579459914(553) ack 351140815 win 2052 <nop,nop,timestamp 239187903 5203
48>
   7: 07:04:04.439827       49.37.41.50.50479 > 20.157.64.169.80: . ack 2880872723 win 2052 <nop,nop,timestamp 239187903 520348>
   8: 07:04:04.440055       49.37.41.50.50479 > 20.157.64.169.80: P 2274994640:2274995193(553) ack 2880872723 win 2052 <nop,nop,timestamp 239187903 520
348>
```

Troubleshooting Tips

- If you see no traffic, check
  - GWLB configuration
  - Interface configuration on the firewa
- If you see the packet only once
  - check your access policy



Agenda

- Introduction
- Azure Load Balancers
- Load Balancer Challenges
- Traffic Flow in Azure
- Gateway Load Balancer Components
- Configuration
- Demo
- Automation and Auto scale Solution Overvi
- Key Takeaway

FMC REST API Support

FTD Configuration Automation

REST API Support the following operations enabling FTD Configuration Automation

- Onboard FTD devices to FMC
- Configure Interfaces
- Create VTEP
- Create VNI Interface

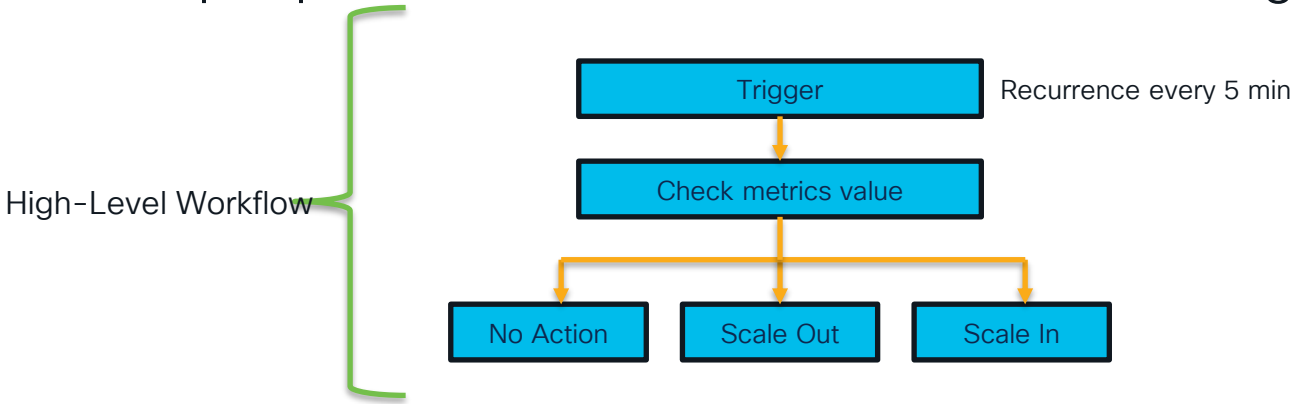
Autoscale Solution with Azure GWLB - Overview

- A serverless solution to scale-out or scale-in firewall instances based on usage
- Helps saving up on resources
- Deploy Resources in Azure using ARM Template
- Uses Function App and Logic App to automate firewall instance scaling
- The Threat defence instances will be part of a **virtual machine scale set**
  - Enable Scaling and managing of the firewall instances
  - Provides high availability of instances
  - collects CPU metrics from the instances

Autoscale Solution with Azure GWLB

Logic App

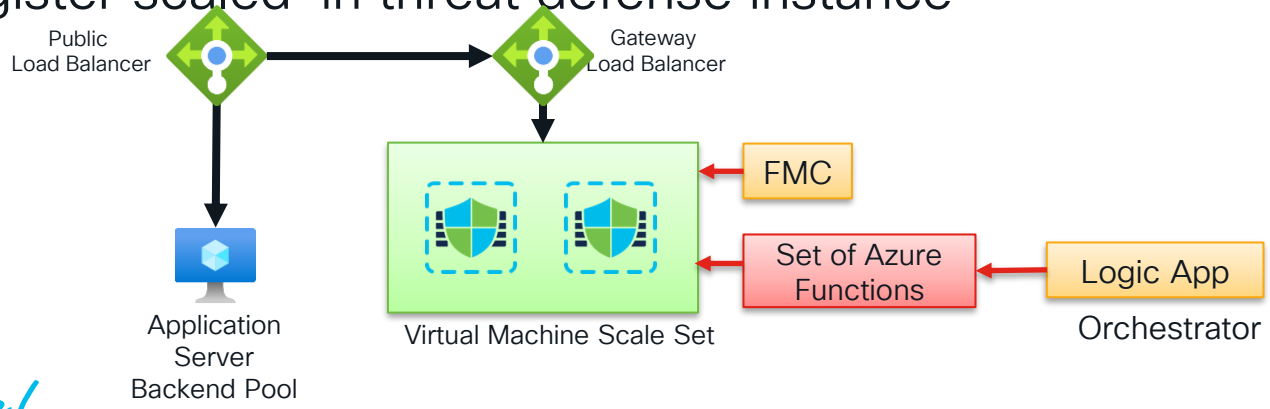
- Create and run automated workflows
- Sequences execution of functions and exchange information between them
- Each step represents an Azure function or built-in logic



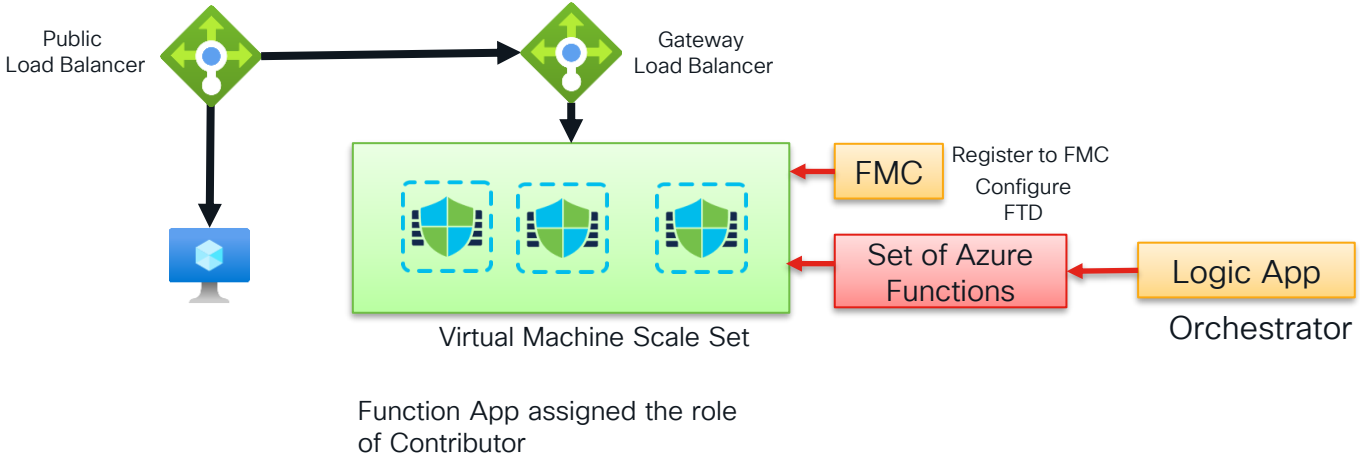
Autoscale Solution with Azure GWLB

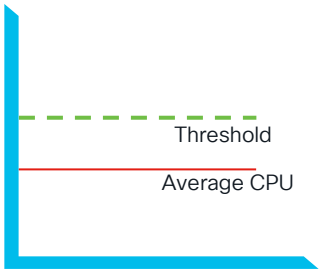
Function App

- Source code written using C++ is compiled and uploaded to the function app
- Probe metrics periodically and trigger scale-in/scale-out operations
- Register and Configure the new threat defense instance
- Unregister scaled-in threat defense instance



Autoscale Solution with Azure GWLB





Wrap Up

Please Fill Out The Survey!

Key Takeaways

- Transparent insertion of firewalls allows a simpler design and minimizes the need for an architectural change.
- This solution simplifies the deployment, management and scaling of the firewall in the Azure environment.
- This solution enables traffic visibility at the endpoint with the original source IP address, which is a requirement for many use cases.

Continue Your Education

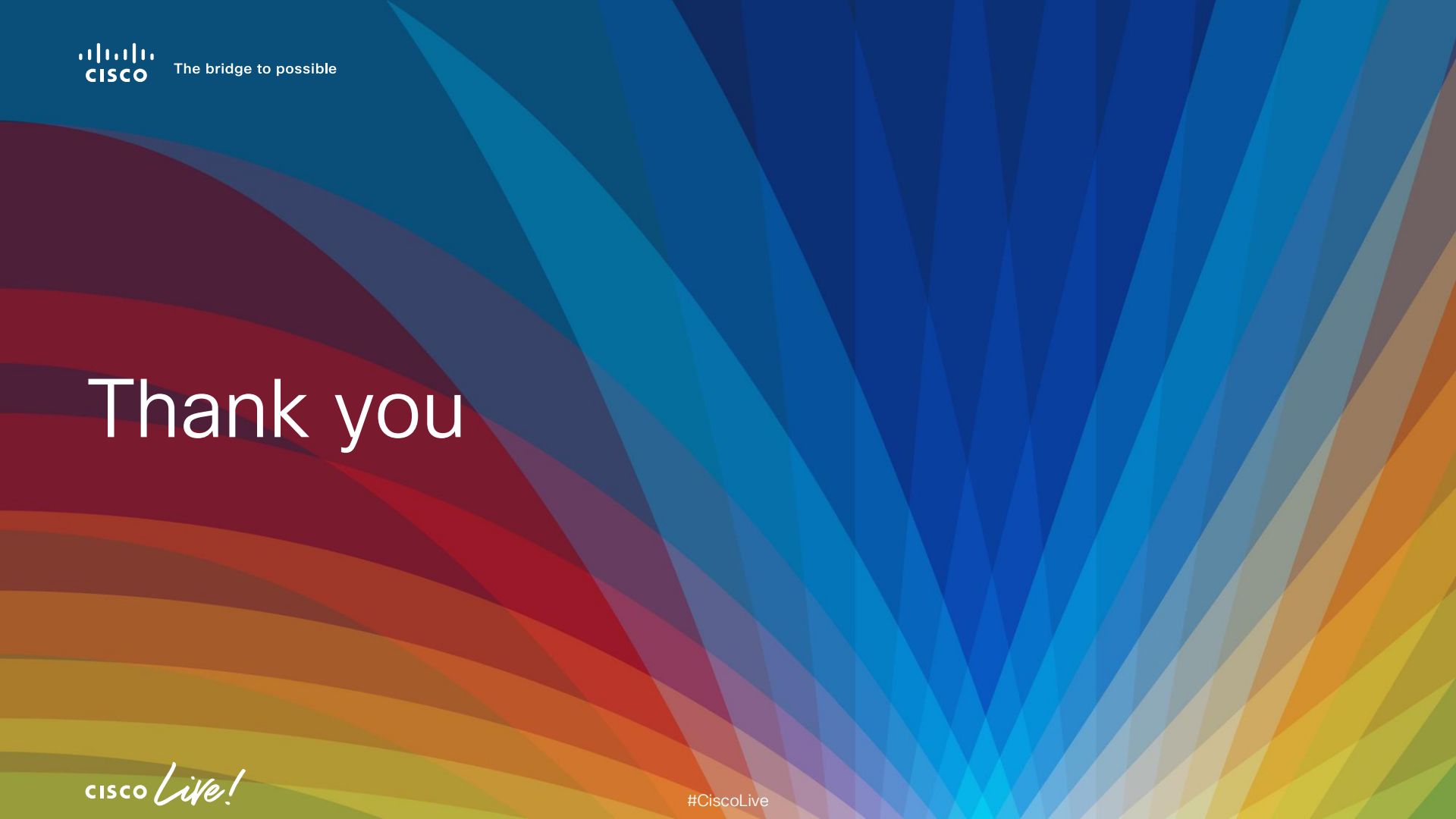
Visit the Cisco Showcase for related demos.

Book your one-on-one Meet the Engineer meeting.

Attend any of the related sessions at the DevNet, Capture the Flag, and Walk-in Labs zones.

Visit the On-Demand Library for more sessions at ciscolive.com/on-demand.

cisco Live!


cisco
The bridge to possible
Thank you
cisco Live!
#CiscoLive


cisco Live!
Let's go
#CiscoLive