

TECHDOCS

PAN-OS® Networking Administrator's Guide

Version 10.2

Contact Information

Corporate Headquarters:
Palo Alto Networks
3000 Tannery Way
Santa Clara, CA 95054
www.paloaltonetworks.com/company/contact-support

About the Documentation

- For the most recent version of this guide or for access to related documentation, visit the Technical Documentation portal docs.paloaltonetworks.com.
- To search for a specific topic, go to our search page docs.paloaltonetworks.com/search.html.
- Have feedback or questions for us? Leave a comment on any page in the portal, or write to us at documentation@paloaltonetworks.com.

Copyright

Palo Alto Networks, Inc.
www.paloaltonetworks.com

© 2021-2025 Palo Alto Networks, Inc. Palo Alto Networks is a registered trademark of Palo Alto Networks. A list of our trademarks can be found at www.paloaltonetworks.com/company/trademarks.html. All other marks mentioned herein may be trademarks of their respective companies.

Last Revised

March 26, 2025

Table of Contents

Networking.....	11
Networking Introduction.....	12
Configure Interfaces.....	15
Tap Interfaces.....	16
Virtual Wire Interfaces.....	18
Layer 2 and Layer 3 Packets over a Virtual Wire.....	19
Port Speeds of Virtual Wire Interfaces.....	20
LLDP over a Virtual Wire.....	20
Aggregated Interfaces for a Virtual Wire.....	20
Virtual Wire Support of High Availability.....	20
Zone Protection for a Virtual Wire Interface.....	21
VLAN-Tagged Traffic.....	21
Virtual Wire Subinterfaces.....	21
Configure Virtual Wires.....	24
Layer 2 Interfaces.....	27
Layer 2 Interfaces with No VLANs.....	27
Layer 2 Interfaces with VLANs.....	28
Configure a Layer 2 Interface.....	29
Configure a Layer 2 Interface, Subinterface, and VLAN.....	29
Manage Per-VLAN Spanning Tree (PVST+) BPDU Rewrite.....	30
Layer 3 Interfaces.....	34
Configure Layer 3 Interfaces.....	34
Manage IPv6 Hosts Using NDP.....	41
Configure an Aggregate Interface Group.....	47
Configure Bonjour Reflector for Network Segmentation.....	51
Use Interface Management Profiles to Restrict Access.....	54
Virtual Routers.....	57
Virtual Router Overview.....	58
Configure Virtual Routers.....	59
Service Routes.....	61
Service Routes Overview.....	62
Configure Service Routes.....	63
Static Routes.....	65
Static Route Overview.....	66
Static Route Removal Based on Path Monitoring.....	67

Table of Contents

Configure a Static Route.....	70
Configure Path Monitoring for a Static Route.....	73
RIP.....	77
RIP Overview.....	78
Configure RIP.....	79
OSPF.....	81
OSPF Concepts.....	82
OSPFv3.....	82
OSPF Neighbors.....	83
OSPF Areas.....	83
OSPF Router Types.....	83
Configure OSPF.....	85
Configure OSPFv3.....	88
Configure OSPF Graceful Restart.....	92
Confirm OSPF Operation.....	93
View the Routing Table.....	93
Confirm OSPF Adjacencies.....	93
Confirm that OSPF Connections are Established.....	93
BGP.....	95
BGP Overview.....	96
MP-BGP.....	97
Configure BGP.....	99
Configure a BGP Peer with MP-BGP for IPv4 or IPv6 Unicast.....	106
Configure a BGP Peer with MP-BGP for IPv4 Multicast.....	109
BGP Confederations.....	111
IP Multicast.....	117
IGMP.....	118
PIM.....	120
Shortest-Path Tree (SPT) and Shared Tree.....	122
PIM Assert Mechanism.....	124
Reverse-Path Forwarding.....	124
Configure IP Multicast.....	126
View IP Multicast Information.....	134
Route Redistribution.....	137
Route Redistribution Overview.....	138
Configure Route Redistribution.....	139
GRE Tunnels.....	143

Table of Contents

GRE Tunnel Overview.....	144
Create a GRE Tunnel.....	146
DHCP.....	149
DHCP Overview.....	150
Firewall as a DHCP Server and Client.....	151
DHCP Messages.....	152
DHCP Addressing.....	154
DHCP Address Allocation Methods.....	154
DHCP Leases.....	154
DHCP Options.....	156
Predefined DHCP Options.....	156
Multiple Values for a DHCP Option.....	157
DHCP Options 43, 55, and 60 and Other Customized Options.....	157
Configure an Interface as a DHCP Server.....	159
Configure an Interface as a DHCP Client.....	163
Configure the Management Interface as a DHCP Client.....	165
Configure an Interface as a DHCP Relay Agent.....	168
Monitor and Troubleshoot DHCP.....	169
View DHCP Server Information.....	169
Clear DHCP Leases.....	169
View DHCP Client Information.....	170
Gather Debug Output about DHCP.....	170
DNS.....	171
DNS Overview.....	172
DNS Proxy Object.....	174
DNS Server Profile.....	175
Multi-Tenant DNS Deployments.....	176
Configure a DNS Proxy Object.....	178
Configure a DNS Server Profile.....	181
Use Case 1: Firewall Requires DNS Resolution.....	182
Use Case 2: ISP Tenant Uses DNS Proxy to Handle DNS Resolution for Security Policies, Reporting, and Services within its Virtual System.....	184
Use Case 3: Firewall Acts as DNS Proxy Between Client and Server.....	187
DNS Proxy Rule and FQDN Matching.....	189
DDNS.....	193
Dynamic DNS Overview.....	194
Configure Dynamic DNS for Firewall Interfaces.....	197
NAT.....	201

Table of Contents

NAT Policy Rules.....	202
NAT Policy Overview.....	202
NAT Address Pools Identified as Address Objects.....	203
Proxy ARP for NAT Address Pools.....	203
Source NAT and Destination NAT.....	206
Source NAT.....	206
Destination NAT.....	208
Destination NAT with DNS Rewrite Use Cases.....	210
NAT Rule Capacities.....	216
Dynamic IP and Port NAT Oversubscription.....	217
Dataplane NAT Memory Statistics.....	218
Configure NAT.....	219
Translate Internal Client IP Addresses to Your Public IP Address (Source DIPP NAT).....	220
Enable Clients on the Internal Network to Access your Public Servers (Destination U-Turn NAT).....	221
Enable Bi-Directional Address Translation for Your Public-Facing Servers (Static Source NAT).....	222
Configure Destination NAT with DNS Rewrite.....	223
Configure Destination NAT Using Dynamic IP Addresses.....	224
Modify the Oversubscription Rate for DIPP NAT.....	226
Reserve Dynamic IP NAT Addresses.....	226
Disable NAT for a Specific Host or Interface.....	228
NAT Configuration Examples.....	229
Destination NAT Example—One-to-One Mapping.....	229
Destination NAT with Port Translation Example.....	230
Destination NAT Example—One-to-Many Mapping.....	231
Source and Destination NAT Example.....	231
Virtual Wire Source NAT Example.....	233
Virtual Wire Static NAT Example.....	234
Virtual Wire Destination NAT Example.....	234
NPTv6.....	237
NPTv6 Overview.....	238
Unique Local Addresses.....	238
Reasons to Use NPTv6.....	239
How NPTv6 Works.....	240
Checksum-Neutral Mapping.....	241
Bi-Directional Translation.....	241
NPTv6 Applied to a Specific Service.....	241
NDP Proxy.....	242

Table of Contents

NPTv6 and NDP Proxy Example.....	244
The ND Cache in NPTv6 Example.....	244
The NDP Proxy in NPTv6 Example.....	244
The NPTv6 Translation in NPTv6 Example.....	245
Neighbors in the ND Cache are Not Translated.....	245
Create an NPTv6 Policy.....	246
NAT64.....	249
NAT64 Overview.....	250
IPv4-Embedded IPv6 Address.....	251
DNS64 Server.....	252
Path MTU Discovery.....	253
IPv6-Initiated Communication.....	254
Configure NAT64 for IPv6-Initiated Communication.....	256
Configure NAT64 for IPv4-Initiated Communication.....	260
Configure NAT64 for IPv4-Initiated Communication with Port Translation.....	263
ECMP.....	267
ECMP Load-Balancing Algorithms.....	268
Configure ECMP on a Virtual Router.....	270
Enable ECMP for Multiple BGP Autonomous Systems.....	273
Verify ECMP.....	274
LLDP.....	275
LLDP Overview.....	276
Supported TLVs in LLDP.....	277
LLDP Syslog Messages and SNMP Traps.....	279
Configure LLDP.....	280
View LLDP Settings and Status.....	282
Clear LLDP Statistics.....	284
BFD.....	285
BFD Overview.....	286
BFD Model, Interface, and Client Support.....	287
Non-Supported RFC Components of BFD.....	287
BFD for Static Routes.....	287
BFD for Dynamic Routing Protocols.....	288
Configure BFD.....	289
Reference: BFD Details.....	296
Session Settings and Timeouts.....	301
Transport Layer Sessions.....	302

Table of Contents

TCP.....	303
TCP Half Closed and TCP Time Wait Timers.....	303
Unverified RST Timer.....	304
TCP Split Handshake Drop.....	305
Maximum Segment Size (MSS).....	306
UDP.....	308
ICMP.....	309
Security Policy Rules Based on ICMP and ICMPv6 Packets.....	309
ICMPv6 Rate Limiting.....	310
Control Specific ICMP or ICMPv6 Types and Codes.....	311
Configure Session Timeouts.....	312
Configure Session Settings.....	315
Session Distribution Policies.....	319
Session Distribution Policy Descriptions.....	319
Change the Session Distribution Policy and View Statistics.....	321
Prevent TCP Split Handshake Session Establishment.....	323
Tunnel Content Inspection.....	325
Tunnel Content Inspection Overview.....	326
Configure Tunnel Content Inspection.....	330
View Inspected Tunnel Activity.....	337
View Tunnel Information in Logs.....	338
Create a Custom Report Based on Tagged Tunnel Traffic.....	339
Tunnel Acceleration Behavior.....	340
Disable Tunnel Acceleration.....	342
Network Packet Broker.....	343
Network Packet Broker Overview.....	344
How Network Packet Broker Works.....	347
Prepare to Deploy Network Packet Broker.....	349
Configure Transparent Bridge Security Chains.....	351
Configure Routed Layer 3 Security Chains.....	357
Network Packet Broker HA Support.....	363
User Interface Changes for Network Packet Broker.....	364
Limitations of Network Packet Broker.....	366
Troubleshoot Network Packet Broker.....	368
Advanced Routing.....	369
Enable Advanced Routing.....	371
Logical Router Overview.....	378
Configure a Logical Router.....	379
Create a Static Route.....	383

Table of Contents

Configure BGP on an Advanced Routing Engine.....	388
Create BGP Routing Profiles.....	402
Create Filters for the Advanced Routing Engine.....	415
Configure OSPFv2 on an Advanced Routing Engine.....	437
Create OSPF Routing Profiles.....	446
Configure OSPFv3 on an Advanced Routing Engine.....	452
Create OSPFv3 Routing Profiles.....	462
Configure RIPv2 on an Advanced Routing Engine.....	468
Create RIPv2 Routing Profiles.....	471
Create BFD Profiles.....	475
Configure IPv4 Multicast.....	477
Create Multicast Routing Profiles.....	487
Create an IPv4 MRoute.....	489

Table of Contents

Networking

All Palo Alto Networks® next-generation firewalls provide a flexible networking architecture that includes support for dynamic routing, switching, and VPN connectivity, and enables you to deploy the firewall into nearly any networking environment.

- [Networking Introduction](#)

Networking Introduction

Networking is the fundamental building block of the firewalls because they must be able to receive data, process it, and forward it. When configuring the Ethernet ports on your firewall, you can choose from tap, virtual wire, Layer2, Layer 3, or AE interface deployments. In addition, to allow you to integrate into a variety of network segments, you can configure different types of interfaces on different ports.

To begin networking, you should first access the Getting Started topic in the PAN-OS® Administrator's Guide. There you learn about segmenting your network and you [Configure Interfaces and Zones](#); that initial task illustrates how to configure Layer 3 interfaces to connect to the internet, your internal network, and your data center applications.

This PAN-OS Networking Administrator's Guide elaborates on that information with topics on how to configure tap, virtual wire, Layer 2, Layer 3, and AE interfaces. After your network interfaces have been configured, you can [Export Configuration Table Data](#) as a PDF or CSV for internal review or audits.

This guide also explains how the firewall supports multiple virtual routers to obtain Layer 3 routes to other subnets and to maintain separate sets of routes. The remaining chapters describe static routes, dynamic routing protocols, and the major features that support networking on the firewall.



You may decide to enable [Advanced Routing](#). The Advanced Routing Engine uses [logical routers](#) instead of virtual routers.

- [Configure Interfaces](#)
- [Virtual Routers](#)
- [Service Routes](#)
- [Static Routes](#)
- [RIP](#)
- [OSPF](#)
- [BGP](#)
- [IP Multicast](#)
- [Route Redistribution](#)
- [GRE Tunnels](#)
- [DHCP](#)
- [DNS](#)
- [DDNS](#)
- [NAT](#)
- [NPTv6](#)
- [NAT64](#)
- [ECMP](#)
- [LLDP](#)

- BFD
- Session Settings and Timeouts
- Tunnel Content Inspection
- Network Packet Broker

Configure Interfaces

A Palo Alto Networks® next-generation firewall can operate in multiple deployments at once because the deployments occur at the interface level. For example, you can configure some interfaces for Layer 3 interfaces to integrate the firewall into your dynamic routing environment, while configuring other interfaces to integrate into your Layer 2 switching network. The following topics describe each type of interface deployment and how to configure it, how to configure Bonjour Reflector, and how to use interface management profiles.

- [Tap Interfaces](#)
- [Virtual Wire Interfaces](#)
- [Layer 2 Interfaces](#)
- [Layer 3 Interfaces](#)
- [Configure an Aggregate Interface Group](#)
- [Configure Bonjour Reflector for Network Segmentation](#)
- [Use Interface Management Profiles to Restrict Access](#)

Tap Interfaces

A network tap is a device that provides a way to access data flowing across a computer network. Tap mode deployment allows you to passively monitor traffic flows across a network by way of a switch SPAN or mirror port.

The SPAN or mirror port permits the copying of traffic from other ports on the switch. By dedicating an interface on the firewall as a tap mode interface and connecting it with a switch SPAN port, the switch SPAN port provides the firewall with the mirrored traffic. This provides application visibility within the network without being in the flow of network traffic.

By deploying the firewall in tap mode, you can get visibility into what applications are running on your network without having to make any changes to your network design. In addition, when in tap mode, the firewall can also identify threats on your network. Keep in mind, however, because the traffic is not running through the firewall when in tap mode it cannot take any action on the traffic, such as blocking traffic with threats or applying QoS traffic control.

To configure a tap interface and begin monitoring the applications and threats on your network:

STEP 1 | Decide which port you want to use as your tap interface and connect it to a switch configured with SPAN/RSPAN or port mirroring.

You will send your network traffic from the SPAN destination port through the firewall so you can have visibility into the applications and threats on your network.

STEP 2 | From the firewall web interface, configure the interface you want to use as your network tap.

1. Select **Network > Interfaces** and select the interface that corresponds to the port you just cabled.
2. Select **Tap** as the **Interface Type**.
3. On the **Config** tab, expand the **Security Zone** and select **New Zone**.
4. In the Zone dialog, enter a **Name** for new zone, for example TapZone, and then click **OK**.

STEP 3 | (Optional) Create any forwarding profiles you want to use.

- [Configure Log Forwarding](#).
- [Configure Syslog Monitoring](#).

STEP 4 | Create **Security Profiles** to scan your network traffic for threats:

1. Select **Objects > Security Profiles**.
2. For each security profile type, **Add** a new profile and set the action to **Alert**.

Because the firewall is not inline with the traffic you cannot use any block or reset actions. By setting the action to alert, you will be able to see any threats the firewall detects in the logs and ACC.

STEP 5 | Create a security policy rule to allow the traffic through the tap interface.

When creating a security policy rule for tap mode, both the source zone and destination zone must be the same.

1. Select **Policies > Security** and click **Add**.
2. In the **Source** tab, set the **Source Zone** to the TapZone you just created.
3. In the **Destination** tab, set the **Destination Zone** to the TapZone also.
4. Set the all of the rule match criteria (**Applications**, **User**, **Service**, **Address**) to **any**.
5. In the **Actions** tab, set the **Action Setting** to **Allow**.
6. Set **Profile Type** to **Profiles** and select each of the security profiles you created to alert you of threats.
7. Verify that **Log at Session End** is enabled.
8. Click **OK**.
9. Place the rule at the top of your rulebase.

STEP 6 | Commit the configuration.

STEP 7 | Monitor the firewall logs (**Monitor > Logs**) and the ACC for insight into the applications and threats on your network.

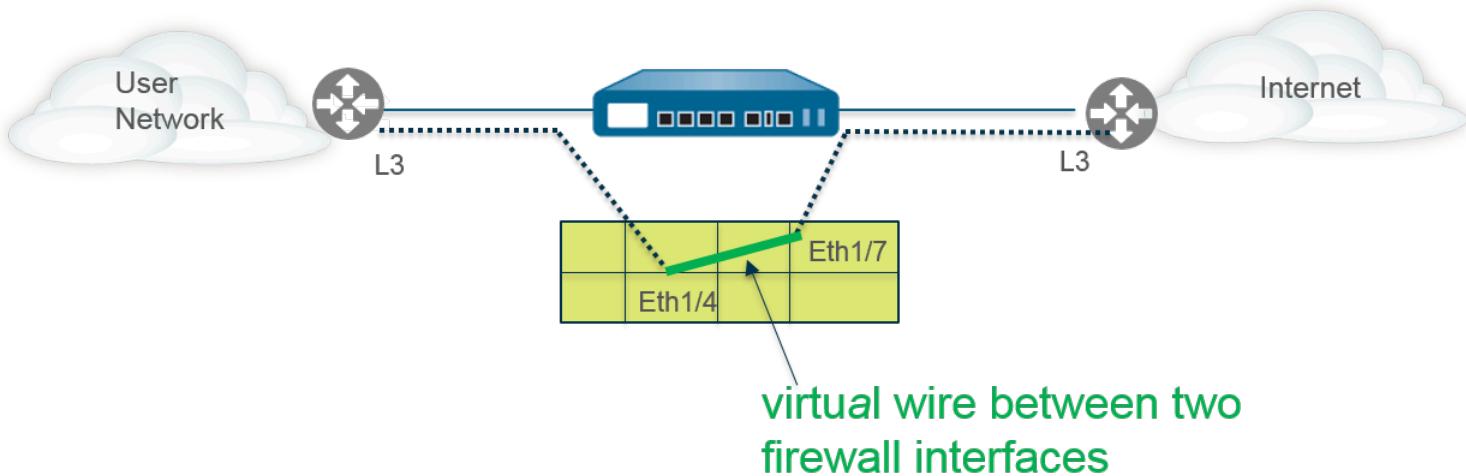
Virtual Wire Interfaces

In a virtual wire deployment, you install a firewall transparently on a network segment by binding two firewall ports (interfaces) together. The virtual wire logically connects the two interfaces; hence, the virtual wire is internal to the firewall.

Use a virtual wire deployment only when you want to seamlessly integrate a firewall into a topology and the two connected interfaces on the firewall don't need to do any switching or routing. For these two interfaces, the firewall is considered a *bump in the wire*.

A virtual wire deployment simplifies firewall installation and configuration because you can insert the firewall into an existing topology without assigning MAC or IP addresses to the interfaces, redesigning the network, or reconfiguring surrounding network devices. The virtual wire supports blocking or allowing traffic based on virtual LAN (VLAN) tags, in addition to supporting security policy rules, App-ID, Content-ID, User-ID, decryption, LLDP, active/passive and active/active HA, QoS, zone protection (with some exceptions), non-IP protocol protection, DoS protection, packet buffer protection, tunnel content inspection, and NAT.

Virtual Wire Deployment (No routing or switching performed by virtual wire interfaces)



Each virtual wire interface is directly connected to a Layer 2 or Layer 3 networking device or host. The virtual wire interfaces have no Layer 2 or Layer 3 addresses. When one of the virtual wire interfaces receives a frame or packet, it ignores any Layer 2 or Layer 3 addresses for switching or routing purposes, but applies your security or NAT policy rules before passing an allowed frame or packet over the virtual wire to the second interface and on to the network device connected to it.

You wouldn't use a virtual wire deployment for interfaces that need to support switching, VPN tunnels, or routing because they require a Layer 2 or Layer 3 address. A virtual wire interface doesn't use an interface management profile, which controls services such as HTTP and ping and therefore requires the interface have an IP address.

All firewalls shipped from the factory have two Ethernet ports (ports 1 and 2) preconfigured as virtual wire interfaces, and these interfaces allow all untagged traffic.



If you're using security group tags (SGTs) in a Cisco TrustSec network, it's a best practice to deploy inline firewalls in either Layer 2 or virtual wire mode. Firewalls in Layer 2 or virtual wire mode can inspect and provide threat prevention for the tagged traffic.



If you don't intend to use the preconfigured virtual wire, you must delete that configuration to prevent it from interfering with other settings you configure on the firewall. See [Set Up Network Access for External Services](#).

- [Layer 2 and Layer 3 Packets over a Virtual Wire](#)
- [Port Speeds of Virtual Wire Interfaces](#)
- [LLDP over a Virtual Wire](#)
- [Aggregated Interfaces for a Virtual Wire](#)
- [Virtual Wire Support of High Availability](#)
- [Zone Protection for a Virtual Wire Interface](#)
- [VLAN-Tagged Traffic](#)
- [Virtual Wire Subinterfaces](#)
- [Configure Virtual Wires](#)

Layer 2 and Layer 3 Packets over a Virtual Wire

A virtual wire interface will allow Layer 2 and Layer 3 packets from connected devices to pass transparently as long as the policies applied to the zone or interface allow the traffic. The virtual wire interfaces themselves don't participate in routing or switching.

For example, the firewall doesn't decrement the TTL in a traceroute packet going over the virtual link because the link is transparent and doesn't count as a hop. Packets such as Operations, Administration and Maintenance (OAM) protocol data units (PDUs), for example, don't terminate at the firewall. Thus, the virtual wire allows the firewall to maintain a transparent presence acting as a pass-through link, while still providing security, NAT, and QoS services.

In order for bridge protocol data units (BPDUs) and other Layer 2 control packets (which are typically untagged) to pass through a virtual wire, the interfaces must be attached to a virtual wire object that allows untagged traffic, and that is the default. If the virtual wire object **Tag Allowed** field is empty, the virtual wire allows untagged traffic. (Security policy rules don't apply to Layer 2 packets.)

In order for routing (Layer 3) control packets to pass through a virtual wire, you must apply a security policy rule that allows the traffic to pass through. For example, apply a security policy rule that allows an application such as BGP or OSPF.

If you want to be able to apply security policy rules to a zone for IPv6 traffic arriving at a virtual wire interface on the firewall, enable IPv6 firewalling. Otherwise, IPv6 traffic is forwarded transparently across the wire.

If you enable multicast firewalling for a virtual wire object and apply it to a virtual wire interface, the firewall inspects multicast traffic and forwards it or not, based on security policy rules. If you don't enable multicast firewalling, the firewall simply forwards multicast traffic transparently.

Fragmentation on a virtual wire occurs the same as in other interface deployment modes.

Port Speeds of Virtual Wire Interfaces

Different firewall models provide various numbers of copper and fiber optic ports, which operate at different speeds. A virtual wire can bind two Ethernet ports of the same type (both copper or both fiber optic), or bind a copper port with a fiber optic port. By default, the **Link Speed** of copper ports on the firewall is set to **auto**, which means the firewall automatically negotiates their speed and transmission mode (**Link Duplex**). When you [Configure Virtual Wires](#), you can also select a specific **Link Speed** and **Link Duplex** but the values for these settings must be the same for both ports in any single virtual wire.

LLDP over a Virtual Wire

Virtual wire interfaces can use [LLDP](#) to discover neighboring devices and their capabilities, and LLDP allows neighboring devices to detect the presence of the firewall in the network. LLDP makes troubleshooting easier especially on a virtual wire, where the firewall would typically go undetected by a ping or traceroute passing through the virtual wire. LLDP provides a way for other devices to detect the firewall in the network. Without LLDP, it is practically impossible for network management systems to detect the presence of a firewall through the virtual link.

Aggregated Interfaces for a Virtual Wire

You can [Configure an Aggregate Interface Group](#) of virtual wire interfaces, but virtual wires don't use LACP. If you configure LACP on devices that connect the firewall to other networks, the virtual wire will pass LACP packets transparently without performing LACP functions.

On a virtual wire, the Palo Alto Networks firewall can pass [Cisco LACP traffic](#) only when the links are not aggregated on the firewall. On a virtual wire, if the links are aggregated, then the firewall could forward the packets to the wrong port in Aggregated Ethernet, which will cause LACP not to function between peers.



In order for aggregate interface groups to function properly, ensure all links belonging to the same LACP group on the same side of the virtual wire are assigned to the same zone.

Virtual Wire Support of High Availability

If you configure the firewall to perform path monitoring for [High Availability](#) using a virtual wire path group, the firewall attempts to resolve ARP for the configured destination IP address by sending ARP packets out both of the virtual wire interfaces. The destination IP address that you are monitoring must be on the same subnetwork as one of the devices surrounding the virtual wire.

Virtual wire interfaces support both active/passive and active/active HA. For an active/active HA deployment with a virtual wire, the scanned packets must be returned to the receiving firewall to preserve the forwarding path. Therefore, if a firewall receives a packet that belongs to the session that the peer HA firewall owns, it sends the packet across the HA3 link to the peer.

Configure Interfaces

You can configure the passive firewall in an HA pair to allow peer devices on either side of the firewall to pre-negotiate LLDP and LACP over a virtual wire before an HA failover occurs. Such a configuration for [LACP and LLDP Pre-Negotiation for Active/Passive HA](#) speeds up HA failovers.

Zone Protection for a Virtual Wire Interface

You can apply zone protection to a virtual wire interface, but because virtual wire interfaces don't perform routing, you can't apply [Packet Based Attack Protection](#) to packets coming with a spoofed IP address, nor can you suppress ICMP TTL Expired error packets or ICMP Frag Needed packets.

By default, a virtual wire interface forwards all non-IP traffic it receives. However, you can apply a zone protection profile with [Protocol Protection](#) to block or allow certain non-IP protocol packets between security zones on a virtual wire.

VLAN-Tagged Traffic

Virtual wire interfaces by default allow all untagged traffic. You can, however, use a virtual wire to connect two interfaces and configure either interface to block or allow traffic based on the virtual LAN (VLAN) tags. VLAN tag 0 indicates untagged traffic.

You can also create multiple subinterfaces, add them into different zones, and then classify traffic according to a VLAN tag or a combination of a VLAN tag with IP classifiers (address, range, or subnet) to apply granular policy control for specific VLAN tags or for VLAN tags from a specific source IP address, range, or subnet.

Virtual Wire Subinterfaces

Virtual wire deployments can use virtual wire subinterfaces to separate traffic into zones. Virtual wire subinterfaces provide flexibility in enforcing distinct policies when you need to manage traffic from multiple customer networks. The subinterfaces allow you to separate and classify traffic into different zones (the zones can belong to separate virtual systems, if required) using the following criteria:

- **VLAN tags**—The example in [Virtual Wire Deployment with Subinterfaces \(VLAN Tags only\)](#) shows an ISP using virtual wire subinterfaces with VLAN tags to separate traffic for two different customers.
- **VLAN tags in conjunction with IP classifiers (address, range, or subnet)**—The following example shows an ISP with two separate virtual systems on a firewall that manages traffic from two different customers. On each virtual system, the example illustrates how virtual wire subinterfaces with VLAN tags and IP classifiers are used to classify traffic into separate zones and apply relevant policy for customers from each network.

Virtual Wire Subinterface Workflow

- Configure two Ethernet interfaces as type virtual wire, and assign these interfaces to a virtual wire.
- Create subinterfaces on the parent Virtual Wire to separate CustomerA and CustomerB traffic. Make sure that the VLAN tags defined on each pair of subinterfaces that are

Virtual Wire Subinterface Workflow

configured as virtual wire(s) are identical. This is essential because a virtual wire does not switch VLAN tags.

- Create new subinterfaces and define IP classifiers. This task is optional and only required if you wish to add additional subinterfaces with IP classifiers for further managing traffic from a customer based on the combination of VLAN tags and a specific source IP address, range or subnet.

You can also use IP classifiers for managing untagged traffic. To do so, you must create a sub-interface with the vlan tag “0”, and define subinterface(s) with IP classifiers for managing untagged traffic using IP classifiers.



IP classification may only be used on the subinterfaces associated with one side of the virtual wire. The subinterfaces defined on the corresponding side of the virtual wire must use the same VLAN tag, but must not include an IP classifier.

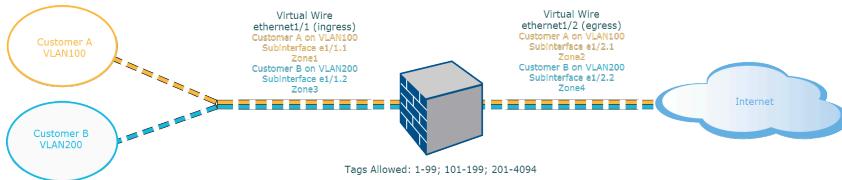


Figure 1: Virtual Wire Deployment with Subinterfaces (VLAN Tags only)

[Virtual Wire Deployment with Subinterfaces \(VLAN Tags only\)](#) depicts CustomerA and CustomerB connected to the firewall through one physical interface, ethernet1/1, configured as a Virtual Wire; it is the ingress interface. A second physical interface, ethernet1/2, is also part of the Virtual Wire; it is the egress interface that provides access to the internet.

For CustomerA, you also have subinterfaces ethernet1/1.1 (ingress) and ethernet1/2.1 (egress). For CustomerB, you have the subinterface ethernet1/1.2 (ingress) and ethernet1/2.2 (egress). When configuring the subinterfaces, you must assign the appropriate VLAN tag and zone in order to apply policies for each customer. In this example, the policies for CustomerA are created between Zone1 and Zone2, and policies for CustomerB are created between Zone3 and Zone4.

When traffic enters the firewall from CustomerA or CustomerB, the VLAN tag on the incoming packet is first matched against the VLAN tag defined on the ingress subinterfaces. In this example, a single subinterface matches the VLAN tag on the incoming packet, hence that subinterface is selected. The policies defined for the zone are evaluated and applied before the packet exits from the corresponding subinterface.



The same VLAN tag must not be defined on the parent virtual wire interface and the subinterface. Verify that the VLAN tags defined on the Tag Allowed list of the parent virtual wire interface ([Network > Virtual Wires](#)) are not included on a subinterface.

[Virtual Wire Deployment with Subinterfaces \(VLAN Tags and IP Classifiers\)](#) depicts CustomerA and CustomerB connected to one physical firewall that has two virtual systems (vsys), in addition to the default virtual system (vsys1). Each virtual system is an independent virtual firewall that

Configure Interfaces

is managed separately for each customer. Each vsys has attached interfaces/subinterfaces and security zones that are managed independently.

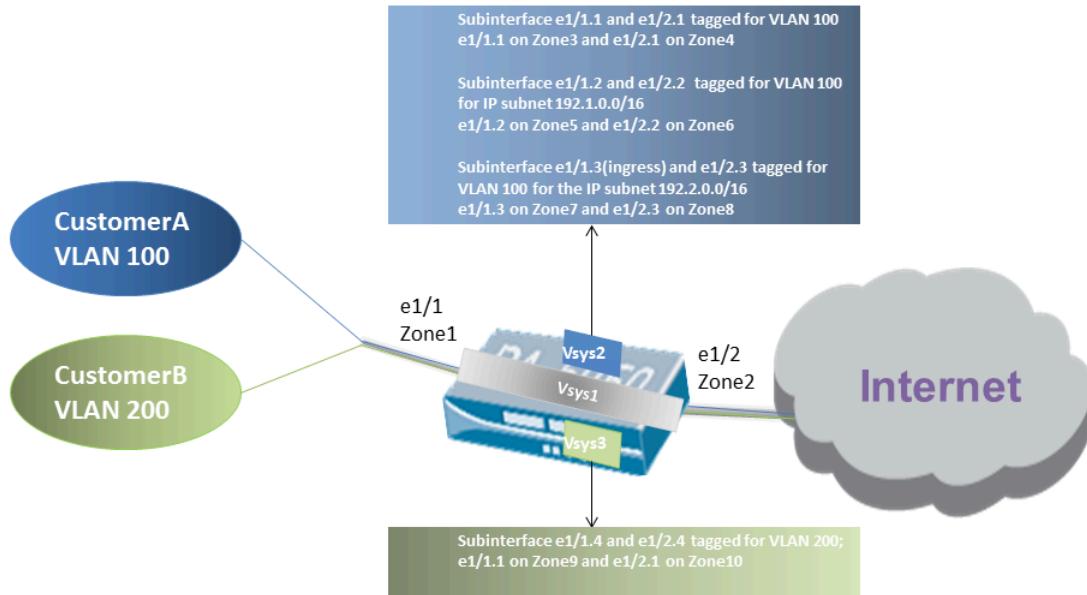


Figure 2: Virtual Wire Deployment with Subinterfaces (VLAN Tags and IP Classifiers)

Vsys1 is set up to use the physical interfaces ethernet1/1 and ethernet1/2 as a virtual wire; ethernet1/1 is the ingress interface and ethernet1/2 is the egress interface that provides access to the Internet. This virtual wire is configured to accept all tagged and untagged traffic with the exception of VLAN tags 100 and 200 that are assigned to the subinterfaces.

CustomerA is managed on vsys2 and CustomerB is managed on vsys3. On vsys2 and vsys3, the following vwire subinterfaces are created with the appropriate VLAN tags and zones to enforce policy measures.

Customer	Vsys	Vwire Subinterfaces	Zone	VLAN Tag	IP Classifier
A	2	e1/1.1 (ingress)	Zone3	100	None
		e1/2.1 (egress)	Zone4	100	
	2	e1/1.2 (ingress)	Zone5	100	IP subnet 192.1.0.0/16
		e1/2.2 (egress)	Zone6	100	
	2	e1/1.3 (ingress)	Zone7	100	IP subnet

Customer	Vsys	Vwire Subinterfaces	Zone	VLAN Tag	IP Classifier
		e1/2.3 (egress)	Zone8	100	192.2.0.0/16
B	3	e1/1.4 (ingress)	Zone9	200	None
		e1/2.4 (egress)	Zone10	200	

When traffic enters the firewall from CustomerA or CustomerB, the VLAN tag on the incoming packet is first matched against the VLAN tag defined on the ingress subinterfaces. In this case, for CustomerA, there are multiple subinterfaces that use the same VLAN tag. Hence, the firewall first narrows the classification to a subinterface based on the source IP address in the packet. The policies defined for the zone are evaluated and applied before the packet exits from the corresponding subinterface.

For return-path traffic, the firewall compares the destination IP address as defined in the IP classifier on the customer-facing subinterface and selects the appropriate virtual wire to route traffic through the accurate subinterface.



*The same VLAN tag must not be defined on the parent virtual wire interface and the subinterface. Verify that the VLAN tags defined on the Tag Allowed list of the parent virtual wire interface (**Network > Virtual Wires**) are not included on a subinterface.*

Configure Virtual Wires

The following task shows how to configure two **Virtual Wire Interfaces** (Ethernet 1/3 and Ethernet 1/4 in this example) to create a virtual wire. The two interfaces must have the same **Link Speed** and transmission mode (**Link Duplex**). For example, a full-duplex 1000Mbps copper port matches a full-duplex 1Gbps fiber optic port.

STEP 1 | Create the first virtual wire interface.

1. Select **Network > Interfaces > Ethernet** and select an interface you have cabled (**ethernet1/3** in this example).
2. Set the **Interface Type** to **Virtual Wire**.

STEP 2 | Attach the interface to a virtual wire object.

1. While still on the same Ethernet interface, on the **Config** tab, select **Virtual Wire** and click **New Virtual Wire**.
2. Enter a **Name** for the virtual wire.
3. For **Interface1**, select the interface you just configured (**ethernet1/3**). (Only interfaces configured as virtual wire interfaces appear in the list.)
4. For **Tag Allowed**, enter **0** to indicate untagged traffic (such as BPDUs and other Layer 2 control traffic) is allowed. The absence of a tag implies tag 0. Enter additional allowed tag integers or ranges of tags, separated by commas (default is 0; range is 0 to 4,094).
5. Select **Multicast Firewalling** if you want to be able to apply security policy rules to multicast traffic going across the virtual wire. Otherwise, multicast traffic is transparently forwarded across the virtual wire.
6. Select **Link State Pass Through** so the firewall can function transparently. When the firewall detects a link down state for a link of the virtual wire, it brings down the other interface in the virtual wire pair. Thus, devices on both sides of the firewall see a consistent link state, as if there were no firewall between them. If you don't select this option, link status is not propagated across the virtual wire.
7. Click **OK** to save the virtual wire object.

STEP 3 | Determine the link speed of the virtual wire interface.

1. While still on the same Ethernet interface, select **Advanced** and note or change the **Link Speed**. The port type determines the speed settings available in the list. By default, copper ports are set to **auto** negotiate link speed. Both virtual wire interfaces must have the same link speed.
2. Click **OK** to save the Ethernet interface.

STEP 4 | Configure the second virtual wire interface (**ethernet1/4** in this example) by repeating the preceding steps.

When you select the **Virtual Wire** object you created, the firewall automatically adds the second virtual wire interface as **Interface2**.

STEP 5 | Create a separate security zone for each virtual wire interface.

1. Select **Network > Zones** and **Add** a zone.
2. Enter the **Name** of the zone (such as **internet**).
3. For **Location**, select the virtual system where the zone applies.
4. For **Type**, select **Virtual Wire**.
5. **Add the Interface** that belongs to the zone.
6. Click **OK**.

STEP 6 | (Optional) Create security policy rules to allow Layer 3 traffic to pass through.

To allow Layer 3 traffic across the virtual wire, [Create a Security Policy Rule](#) to allow traffic from the user zone to the internet zone, and another to allow traffic from the internet zone to the user zone, selecting the applications you want to allow, such as BGP or OSPF.

STEP 7 | (Optional) Enable IPv6 firewalling.

If you want to be able to apply security policy rules to IPv6 traffic arriving at the virtual wire interface, enable IPv6 firewalling. Otherwise, IPv6 traffic is forwarded transparently.

1. Select **Device > Setup > Session** and edit Session Settings.
2. Select **Enable IPv6 Firewalling**.
3. Click **OK**.

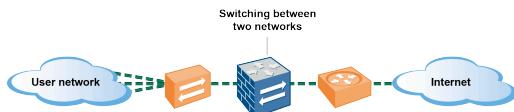
STEP 8 | Commit your changes.

STEP 9 | (Optional) Configure an LLDP profile and apply it to the virtual wire interfaces (see [Configure LLDP](#)).

STEP 10 | (Optional) Apply non-IP protocol control to the virtual wire zones ([Configure Protocol Protection](#)). Otherwise, all non-IP traffic is forwarded over the virtual wire.

Layer 2 Interfaces

In a Layer 2 deployment, the firewall provides switching between two or more networks. Devices are connected to a Layer 2 segment; the firewall forwards the frames to the proper port, which is associated with the MAC address identified in the frame. [Configure a Layer 2 Interface](#) when switching is required.



 If you're using security group tags (SGTs) in a Cisco TrustSec network, it's a best practice to deploy inline firewalls in either Layer 2 or virtual wire mode. Firewalls in Layer 2 or virtual wire mode can inspect and provide threat prevention for the tagged traffic.

The following topics describe the different types of Layer 2 interfaces you can configure for each type of deployment you need, including details on using virtual LANs (VLANs) for traffic and policy separation among groups. Another topic describes how the firewall rewrites the inbound port VLAN ID number in a Cisco per-VLAN spanning tree (PVST+) or Rapid PVST+ bridge protocol data unit (BPDU).

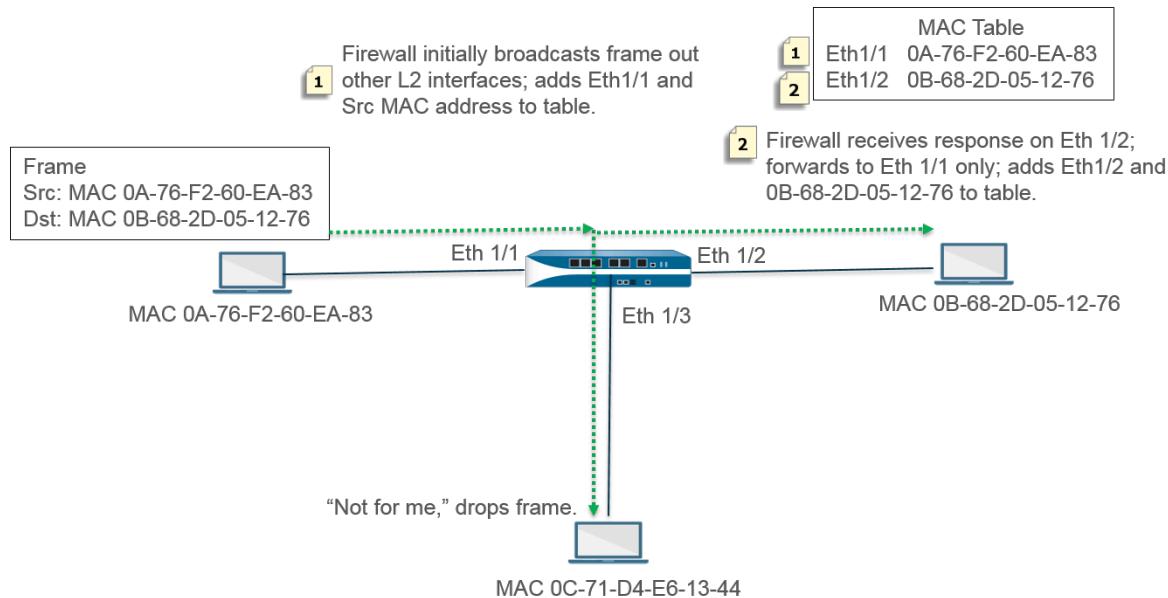
- [Layer 2 Interfaces with No VLANs](#)
- [Layer 2 Interfaces with VLANs](#)
- [Configure a Layer 2 Interface](#)
- [Configure a Layer 2 Interface, Subinterface, and VLAN](#)
- [Manage Per-VLAN Spanning Tree \(PVST+\) BPDU Rewrite](#)

Layer 2 Interfaces with No VLANs

[Configure a Layer 2 Interface](#) on the firewall so it can act as a switch in your layer 2 network (not at the edge of the network). The Layer 2 hosts are probably geographically close to each other and belong to a single broadcast domain. The firewall provides security between the Layer 2 hosts when you assign the interfaces to security zones and apply security rules to the zones.

The hosts communicate with the firewall and each other at Layer 2 of the OSI model by exchanging frames. A frame contains an Ethernet header that includes a source and destination Media Access Control (MAC) address, which is a physical hardware address. MAC addresses are 48-bit hexadecimal numbers formatted as six octets separated by a colon or hyphen (for example, 00-85-7E-46-F1-B2).

The following figure has a firewall with three Layer 2 interfaces that each connect to a Layer 2 host in a one-to-one mapping.



The firewall begins with an empty MAC table. When the host with source address 0A-76-F2-60-EA-83 sends a frame to the firewall, the firewall doesn't have destination address 0B-68-2D-05-12-76 in its MAC table, so it doesn't know which interface to forward the frame to; it broadcasts the frame to all of its Layer 2 interfaces. The firewall puts source address 0A-76-F2-60-EA-83 and associated Eth1/1 into its MAC table.

The host at 0C-71-D4-E6-13-44 receives the broadcast, but the destination MAC address is not its own MAC address, so it drops the frame.

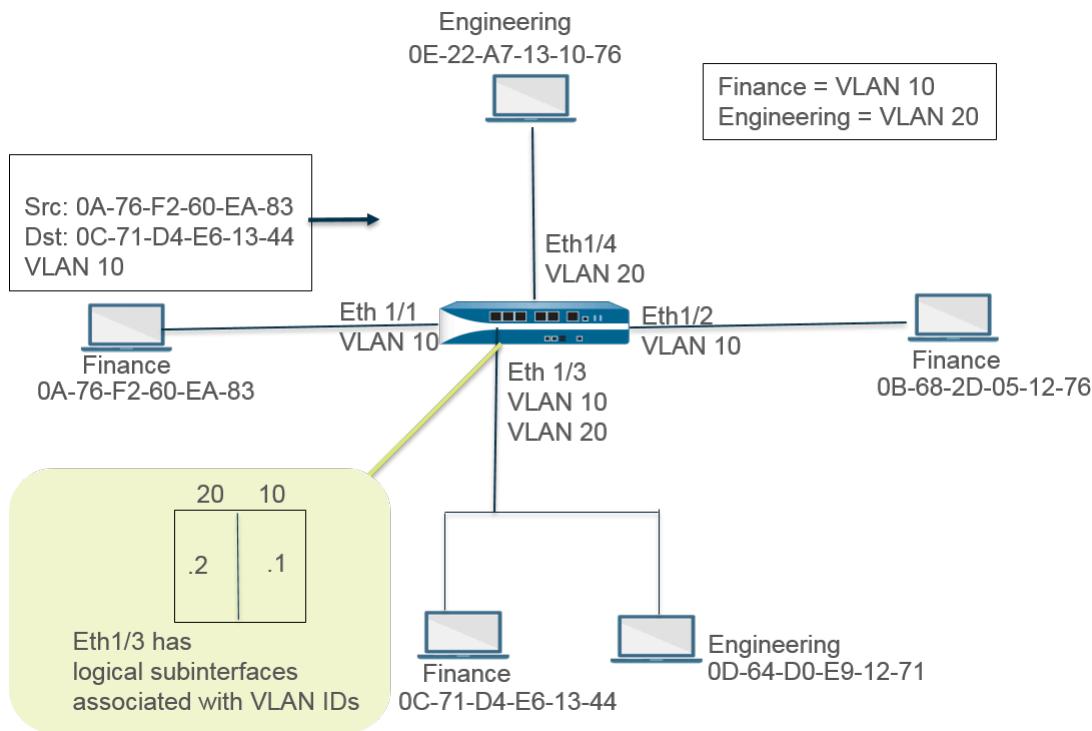
The receiving interface Ethernet 1/2 forwards the frame to its host. When host 0B-68-2D-05-12-76 responds, it uses the destination address 0A-76-F2-60-EA-83, and the firewall adds to its MAC table Ethernet 1/2 as the interface to reach 0B-68-2D-05-12-76.

Layer 2 Interfaces with VLANs

When your organization wants to divide a LAN into separate virtual LANs (VLANs) to keep traffic and policies for different departments separate, you can logically group Layer 2 hosts into VLANs and thus divide a Layer 2 network segment into broadcast domains. For example, you can create VLANs for the Finance and Engineering departments. To do so, [Configure a Layer 2 Interface, Subinterface, and VLAN](#).

The firewall acts as a switch to forward a frame with an Ethernet header containing a VLAN ID, and the destination interface must have a subinterface with that VLAN ID in order to receive that frame and forward it to the host. You configure a Layer 2 interface on the firewall and configure one or more logical subinterfaces for the interface, each with a VLAN tag (ID).

In the following figure, the firewall has four Layer 2 interfaces that connect to Layer 2 hosts belonging to different departments within an organization. Ethernet interface 1/3 is configured with subinterface .1 (tagged with VLAN 10) and subinterface .2 (tagged with VLAN 20), thus there are two broadcast domains on that segment. Hosts in VLAN 10 belong to Finance; hosts in VLAN 20 belong to Engineering.



In this example, the host at MAC address 0A-76-F2-60-EA-83 sends a frame with VLAN ID 10 to the firewall, which the firewall broadcasts to its other L2 interfaces. Ethernet interface 1/3 accepts the frame because it's connected to the host with destination 0C-71-D4-E6-13-44 and its subinterface .1 is assigned VLAN 10. Ethernet interface 1/3 forwards the frame to the Finance host.

Configure a Layer 2 Interface

Configure [Layer 2 Interfaces with No VLANs](#) when you want Layer 2 switching and you don't need to separate traffic among VLANs.

STEP 1 | Configure a Layer 2 interface.

1. Select **Network > Interfaces > Ethernet** and select an interface. The **Interface Name** is fixed, such as `ethernet1/1`.
2. For **Interface Type**, select **Layer2**.
3. Select the **Config** tab and assign the interface to a **Security Zone** or create a **New Zone**.
4. Configure additional Layer 2 interfaces on the firewall that connect to other Layer 2 hosts.

STEP 2 | Commit.

Click **OK** and **Commit**.

Configure a Layer 2 Interface, Subinterface, and VLAN

Configure [Layer 2 Interfaces with VLANs](#) when you want Layer 2 switching and traffic separation among VLANs. You can optionally control non-IP protocols between security zones on a Layer 2 interface or between interfaces within a single zone on a Layer 2 VLAN.

STEP 1 | Configure a Layer 2 interface and subinterface and assign a VLAN ID.

1. Select **Network > Interfaces > Ethernet** and select an interface. The **Interface Name** is fixed, such as `ethernet1/1`.
2. For **Interface Type**, select **Layer2**.
3. Select the **Config** tab.
4. For **VLAN**, leave the setting **None**.
5. Assign the interface to a **Security Zone** or create a **New Zone**.
6. Click **OK**.
7. With the Ethernet interface highlighted, click **Add Subinterface**.
8. The **Interface Name** remains fixed. After the period, enter the subinterface number, in the range 1 to 9,999.
9. Enter a **VLAN Tag ID** in the range 1 to 4,094.
10. Assign the subinterface to a **Security Zone**.
11. Click **OK**.

STEP 2 | Commit.

Click **Commit**.

STEP 3 | (Optional) Apply a Zone Protection profile with protocol protection to control non-IP protocol packets between Layer 2 zones (or between interfaces within a Layer 2 zone).

[Configure Protocol Protection](#).

Manage Per-VLAN Spanning Tree (PVST+) BPDU Rewrite

When an interface on the firewall is configured for a [Layer 2 deployment](#), the firewall rewrites the inbound Port VLAN ID (PVID) number in a Cisco per-VLAN spanning tree (PVST+) or Rapid PVST+ bridge protocol data unit (BPDU) to the proper outbound VLAN ID number and forwards the BPDU out. This default behavior beginning in PAN-OS 7.1 allows the firewall to correctly tag Cisco proprietary PVST+ and Rapid PVST+ frames between Cisco switches in VLANs on either side of the firewall so that spanning tree loop detection using Cisco PVST+ and Rapid PVST+ can function properly. The firewall is not participating in the Spanning Tree Protocol (STP) election process and there is no behavior change for other types of spanning tree.



The Cisco switch must have the `loopguard` disabled for the PVST+ or Rapid PVST+ BPDU rewrite to function properly on the firewall.

This feature is supported on Layer 2 Ethernet and Aggregated Ethernet (AE) interfaces only. The firewall supports a PVID range of 1 to 4,094 with a native VLAN ID of 1 to be compatible with the Cisco native VLAN implementation.

To support the PVST+ BPDU rewrite feature, PAN-OS supports the concept of a PVST+ native VLAN. Frames sent to and received from a native VLAN are untagged with a PVID equal to the native VLAN. All switches and firewalls in the same Layer 2 deployment must have the same native VLAN for PVST+ to function properly. Although the Cisco native VLAN defaults to `vlan1`, the VLAN ID could be a number other than 1.

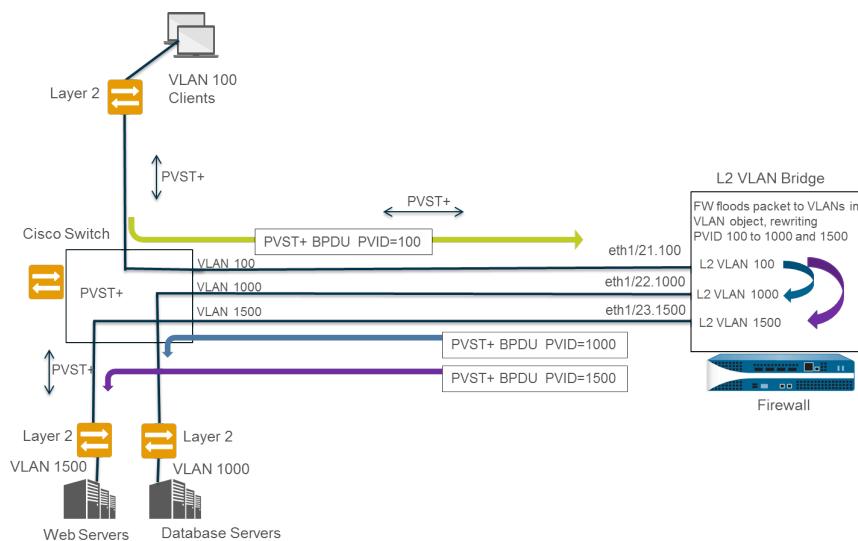
Configure Interfaces

For example, the firewall is configured with a VLAN object (named VLAN_BRIDGE), which describes the interfaces and subinterfaces that belong to a switch or broadcast domain. In this example, the VLAN includes three subinterfaces: ethernet1/21.100 tagged with 100, ethernet1/22.1000, and ethernet1/23.1500 tagged with 1500.

The subinterfaces belonging to VLAN_BRIDGE look like this:

Ethernet	VLAN	Loopback	Tunnel	SD-WAN			
INTERFACE	INTERFACE TYPE	LINK STATE	TAG	VLAN / VIRTUAL-WIRE	SECURITY ZONE	SD-WAN INTERFACE PROFILE	UPSTREAM NAT
ethernet1/21	Layer2	Up	Untagged	none	none		Disabled
ethernet1/21.100	Layer2	Up	100	VLAN_BRIDGE	Zone_Trust		Disabled
ethernet1/22	Layer2	Up	Untagged	none	none		Disabled
ethernet1/22.1000	Layer2	Up	1000	VLAN_BRIDGE	Zone_Untrust		Disabled
ethernet1/23	Layer2	Up	Untagged	none	none		Disabled
ethernet1/23.1500	Layer2	Up	1500	VLAN_BRIDGE	Zone_Management		Disabled

The sequence in which the firewall automatically rewrites the PVST+ BPDU is shown in the following graphic and explanation:



1. The Cisco switch port belonging to VLAN 100 sends a PVST+ BPDU—with the PVID and 802.1Q VLAN tag set to 100—to the firewall.
2. The firewall interfaces and subinterfaces are configured as a Layer 2 interface type. The ingress subinterface on the firewall is tagged with VLAN 100, which matches the PVID and VLAN tag of the incoming BPDU, so the firewall accepts the BPDU. The firewall floods the PVST+ BPDU to all other interfaces belonging to the same VLAN object (in this example, ethernet1/22.1000 and ethernet1/23.1500). If the VLAN tags did not match, the firewall would instead drop the BPDU.
3. When the firewall floods the BPDU out through other interfaces (belonging to the same VLAN object), the firewall rewrites the PVID and any 802.1Q VLAN tags to match the VLAN tag of the egress interface. In this example, the firewall rewrites the BPDU PVID from 100 to 1000.

for one subinterface and from 100 to 1500 for the second subinterface as the BPDU traverses the Layer 2 bridge on the firewall.

4. Each Cisco switch receives the correct PVID and VLAN tag on the incoming BPDU and processes the PVST+ packet to detect possible loops in the network.

The following CLI operational commands allow you to manage PVST+ and Rapid PVST+ BPDUs.

- Globally disable or re-enable the PVST+ and Rapid PVST+ BPDU rewrite of the PVID (default is enabled).

```
set session rewrite-pvst-pvid <yes|no>
```

- Set the native VLAN ID for the firewall (range is 1 to 4,094; default is 1).

 If the native VLAN ID on your switch is a value other than 1, you must set the native VLAN ID on the firewall to that same number; otherwise, the firewall will drop packets with that VLAN ID. This applies to trunked and non-trunked interfaces.

```
set session pvst-native-vlan-id <vid>
```

- Drop all STP BPDU packets.

```
set session drop-stp-packet <yes|no>
```

Examples of why you might want to drop all STP BPDU packets:

- If there is only one switch on each side of the firewall and no other connections between the switches that can cause a loop, then STP is not required and can be disabled on the switch or blocked by the firewall.
 - If there is a misbehaving STP switch inappropriately flooding BPDUs, you can stop the STP packets at the firewall to stop the BPDU flood.
- Verify whether PVST+BPDU rewrite is enabled, view the PVST native VLAN ID, and determine whether the firewall is dropping all STP BPDU packets.

```
show vlan all
```

```
pvst+ tag rewrite: disabled
pvst native vlan id:      5
drop stp:                  disabled
total vlans shown:        1
name      interface          virtual interface
bridge    ethernet1/1
          ethernet1/2
          ethernet1/1.1
          ethernet1/2.1
```

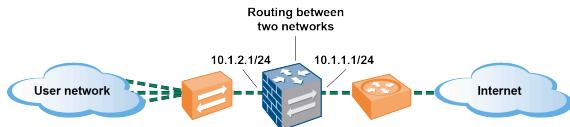
- Troubleshoot PVST+ BPDU errors.

show counter global

Look at the `flow_pvid_inconsistent` counter, which counts the number of times the 802.1Q Tag and PVID fields inside a PVST+ BPDU packet don't match.

Layer 3 Interfaces

In a Layer 3 deployment, the firewall routes traffic between multiple ports. Before you can [Configure Layer 3 Interfaces](#), you must configure the [virtual router](#) that you want the firewall to use to route the traffic for each Layer 3 interface.



If you're using security group tags (SGTs) in a Cisco TrustSec network, it's a best practice to deploy inline firewalls in either Layer 2 or virtual wire mode. However, if you need to use a Layer 3 firewall in a Cisco TrustSec network, you should deploy the Layer 3 firewall between two SGT exchange protocol (SXP) peers, and configure the firewall to allow traffic between the SXP peers.

The following topics describe how to configure Layer 3 interfaces, and how to use Neighbor Discovery Protocol (NDP) to provision IPv6 hosts and view the IPv6 addresses of devices on the link local network to quickly locate devices.

- [Configure Layer 3 Interfaces](#)
- [Manage IPv6 Hosts Using NDP](#)

Configure Layer 3 Interfaces

The following procedure is required to configure [Layer 3 Interfaces](#) (Ethernet, VLAN, loopback, and tunnel interfaces) with IPv4 or IPv6 addresses so that the firewall can perform routing on these interfaces. If a tunnel is used for routing or if tunnel monitoring is turned on, the tunnel needs an IP address. Before performing the following task, define one or more [virtual routers](#) on a legacy routing engine or [logical routers](#) on an Advanced Routing Engine.

You would typically use the following procedure to configure an external interface that connects to the internet and an interface for your internal network. You can configure both IPv4 and IPv6 addresses on a single interface.



PAN-OS firewall models support a maximum of 16,000 IP addresses assigned to physical or virtual Layer 3 interfaces; this maximum includes both IPv4 and IPv6 addresses.

If you're using IPv6 routes, you can configure the firewall to provide [IPv6 Router Advertisements for DNS Configuration](#). The firewall provisions IPv6 DNS clients with Recursive DNS Server (RDNS) addresses and a DNS Search List so that the client can resolve its IPv6 DNS requests. Thus the firewall is acting like a DHCPv6 server for you.

STEP 1 | Select an interface and configure it with a security zone.

1. Select **Network > Interfaces** and either **Ethernet**, **VLAN**, **loopback**, or **Tunnel**, depending on what type of interface you want.
2. Select the interface to configure.
3. Select the **Interface Type—Layer3**.
4. On the **Config** tab, for **Virtual Router**, select the virtual router you are configuring, such as **default**.
5. For **Virtual System**, select the virtual system you are configuring if on a multi-virtual system firewall.
6. For **Security Zone**, select the zone to which the interface belongs or create a **New Zone**.
7. Click **OK**.

STEP 2 | Configure the interface with an IPv4 address.

You can assign an IPv4 address to a Layer 3 interface in one of three ways:

- **Static**
 - **DHCP Client**—The firewall interface acts as a DHCP client and receives a dynamically assigned IP address. The firewall also provides the capability to propagate settings received by the DHCP client interface into a DHCP server operating on the firewall. This is most commonly used to propagate DNS server settings from an Internet service provider to client machines operating on the network protected by the firewall.
 - **PPPoE**—Configure the interface as a Point-to-Point Protocol over Ethernet (PPPoE) termination point to support connectivity in a Digital Subscriber Line (DSL) environment where there is a DSL modem but no other PPPoE device to terminate the connection.
1. Select **Network > Interfaces** and either **Ethernet**, **VLAN**, **loopback**, or **Tunnel**, depending on what type of interface you want.
 2. Select the interface to configure.
 3. To configure the interface with a static IPv4 address, on the **IPv4** tab, set **Type** to **Static**.
 4. Add a **Name** and optional **Description** for the address.
 5. For **Type**, select one of the following:
 - **IP Netmask**—Enter the IP address and network mask to assign to the interface, for example, 208.80.56.100/24.
 -  If you're using a /31 subnet mask for the Layer 3 interface address, the interface must be configured with the .1/31 address in order for utilities such as ping to work properly.
 -  If you're configuring a loopback interface with an IPv4 address, it must have a /32 subnet mask; for example, 192.168.2.1/32.
 - **IP Range**—Enter an IP address range, such as 192.168.2.1-192.168.2.4.
 - **FQDN**—Enter a Fully Qualified Domain Name.
 6. Select **Tags** to apply to the address.
 7. Click **OK**.

STEP 3 | Configure an interface with Point-to-Point Protocol over Ethernet (PPPoE). See [Layer 3 Interfaces](#).

 *PPPoE is not supported in HA active/active mode.*

1. Select **Network > Interfaces** and either **Ethernet**, **VLAN**, **loopback**, or **Tunnel**.
2. Select the interface to configure.
3. On the **IPv4** tab, set **Type** to **PPPoE**.
4. On the **General** tab, select **Enable** to activate the interface for PPPoE termination.
5. Enter the **Username** for the point-to-point connection.
6. Enter the **Password** for the username and **Confirm Password**.
7. Click **OK**.

STEP 4 | Configure an [Interface as a DHCP Client](#) so that it receives a dynamically-assigned IPv4 address.

 *DHCP client is not supported in HA active/active mode.*

STEP 5 | Configure an interface with a static IPv6 address.

1. Select **Network > Interfaces** and either **Ethernet**, **VLAN**, **loopback**, or **Tunnel**.
2. Select the interface to configure.
3. On the **IPv6** tab, select **Enable IPv6 on the interface** to enable IPv6 addressing on the interface.
4. For **Interface ID**, enter the 64-bit extended unique identifier (EUI-64) in hexadecimal format (for example, 00:26:08:FF:FE:DE:4E:29). If you leave this field blank, the firewall uses the EUI-64 generated from the MAC address of the physical interface. If you enable the **Use interface ID as host portion** option when adding an address, the firewall uses the Interface ID as the host portion of that address.
5. Add the **IPv6 Address** or select an address group.
6. Select **Enable address on interface** to enable this IPv6 address on the interface.
7. Select **Use interface ID as host portion** to use the Interface ID as the host portion of the IPv6 address.
8. (**Optional**) Select **Anycast** to make the IPv6 address (route) an Anycast address (route), which means multiple locations can advertise the same prefix, and IPv6 sends the

anycast traffic to the node it considers the nearest, based on routing protocol costs and other factors.

9. **(Ethernet interface only)** Select **Send Router Advertisement (RA)** to enable the firewall to send this address in Router Advertisements, in which case you must also enable the global **Enable Router Advertisement** option on the interface (next step).
10. **(Ethernet interface only)** Enter the **Valid Lifetime (sec)**, in seconds, that the firewall considers the address valid. The Valid Lifetime must equal or exceed the **Preferred Lifetime (sec)** (default is 2,592,000).
11. **(Ethernet interface only)** Enter the **Preferred Lifetime (sec)** (in seconds) that the valid address is preferred, which means the firewall can use it to send and receive traffic. After the Preferred Lifetime expires, the firewall can't use the address to establish new connections, but any existing connections are valid until the **Valid Lifetime** expires (default is 604,800).
12. **(Ethernet interface only)** Select **On-link** if systems that have addresses within the prefix are reachable without a router.
13. **(Ethernet interface only)** Select **Autonomous** if systems can independently create an IP address by combining the advertised prefix with an Interface ID.
14. Click **OK**.

STEP 6 | (Ethernet or VLAN interface using IPv6 address only) Enable the firewall to send IPv6 Router Advertisements (RAs) from an interface, and optionally tune RA parameters.



Tune RA parameters for either of these reasons: To interoperate with a router/host that uses different values. To achieve fast convergence when multiple gateways are present. For example, set lower **Min Interval**, **Max Interval**, and **Router Lifetime** values so the IPv6 client/host can quickly change the default gateway after the primary gateway fails, and start forwarding to another default gateway in the network.

1. Select **Network > Interfaces** and **Ethernet** or **VLAN**.
2. Select the interface you want to configure.
3. Select **IPv6**.
4. Select **Enable IPv6 on the interface**.
5. On the **Router Advertisement** tab, select **Enable Router Advertisement** (default is disabled).
6. (**Optional**) Set **Min Interval (sec)**, the minimum interval, in seconds, between RAs the firewall sends (range is 3-1,350; default is 200). The firewall sends RAs at random intervals between the minimum and maximum values you set.
7. (**Optional**) Set **Max Interval (sec)**, the maximum interval, in seconds, between RAs the firewall sends (range is 4-1,800; default is 600). The firewall sends RAs at random intervals between the minimum and maximum values you set.
8. (**Optional**) Set **Hop Limit** to apply to clients for outgoing packets (range is 1-255; default is 64). Enter 0 for no hop limit.
9. (**Optional**) Set **Link MTU**, the link maximum transmission unit (MTU) to apply to clients (range is 1,280-9,192; default is **unspecified**). Select **unspecified** for no link MTU.
10. (**Optional**) Set **Reachable Time (ms)**, the reachable time, in milliseconds, that the client will use to assume a neighbor is reachable after receiving a Reachability Confirmation message. Select **unspecified** for no reachable time value (range is 0-3,600,000; default is **unspecified**).
11. (**Optional**) Set **Retrans Time (ms)**, the retransmission timer that determines how long the client will wait, in milliseconds, before retransmitting Neighbor Solicitation messages. Select **unspecified** for no retransmission time (range is 0-4,294,967,295; default is **unspecified**).
12. (**Optional**) Set **Router Lifetime (sec)** to specify how long, in seconds, the client will use the firewall as the default gateway (range is 0-9,000; default is 1,800). Zero specifies that the firewall is not the default gateway. When the lifetime expires, the client removes the firewall entry from its Default Router List and uses another router as the default gateway.
13. Set **Router Preference**, which the client uses to select a preferred router if the network segment has multiple IPv6 routers. **High**, **Medium** (default), or **Low** is the priority that

the RA advertises indicating the relative priority of firewall virtual router relative to other routers on the segment.

14. Select **Managed Configuration** to indicate to the client that addresses are available via DHCPv6.
15. Select **Other Configuration** to indicate to the client that other address information (such as DNS-related settings) is available via DHCPv6.
16. Select **Consistency Check** to have the firewall verify that RAs sent from other routers are advertising consistent information on the link. The firewall logs any inconsistencies.
17. Click **OK**.

STEP 7 | (Ethernet or VLAN interface using IPv6 address only) Specify the Recursive DNS Server addresses and DNS Search List the firewall will advertise in ND Router Advertisements from this interface.

The RDNS servers and DNS Search List are part of the DNS configuration for the DNS client so that the client can resolve IPv6 DNS requests.

1. Select **Network > Interfaces** and **Ethernet or VLAN**.
2. Select the interface you are configuring.
3. Select **IPv6 > DNS Support**.
4. **Include DNS information in Router Advertisement** to enable the firewall to send IPv6 DNS information.
5. For **DNS Server**, Add the IPv6 address of a Recursive DNS Server. **Add** up to eight Recursive DNS servers. The firewall sends server addresses in an ICMPv6 Router Advertisement in order from top to bottom.
6. Specify the **Lifetime** in seconds, which is the maximum length of time the client can use the specific RDNS Server to resolve domain names.
 - The **Lifetime** range is any value equal to or between the **Max Interval** (that you configured on the **Router Advertisement** tab) and two times that **Max Interval**. For example, if your Max Interval is 600 seconds, the Lifetime range is 600-1,200 seconds.
 - The default **Lifetime** is 1,200 seconds.
7. For **DNS Suffix**, Add a **DNS Suffix** (domain name of a maximum of 255 bytes). **Add** up to eight DNS suffixes. The firewall sends suffixes in an ICMPv6 Router Advertisement in order from top to bottom.
8. Specify the **Lifetime** in seconds, which is the maximum length of time the client can use the suffix. The Lifetime has the same range and default value as the **Server**.
9. Click **OK**.

STEP 8 | (**Ethernet or VLAN interface**) Specify static ARP entries. Static ARP entries reduce ARP processing.

1. Select **Network > Interfaces** and **Ethernet or VLAN**.
2. Select the interface you are configuring.
3. Select **Advanced > ARP Entries**.
4. Add an **IP Address** and its corresponding **MAC Address** (hardware or media access control address). For a VLAN interface, you must also select the **Interface**.



Static ARP entries do not time out. Auto learned ARP entries in the cache time out in 1,800 seconds by default; you can customize the ARP cache timeout; see [Configure Session Timeouts](#).

5. Click **OK**.

STEP 9 | (**Ethernet or VLAN interface**) Specify static Neighbor Discovery Protocol (NDP) entries. NDP for IPv6 performs functions similar to those provided by ARP for IPv4.

1. Select **Network > Interfaces** and **Ethernet or VLAN**.
2. Select the interface you are configuring.
3. Select **Advanced > ND Entries**.
4. Add an **IPv6 Address** and its corresponding **MAC Address**.
5. Click **OK**.

STEP 10 | (Optional) Enable services on the interface.

1. To enable services on the interface, select **Network > Interfaces** and **Ethernet or VLAN**.
2. Select the interface you are configuring.
3. Select **Advanced > Other Info**.
4. Expand the **Management Profile** list and select a profile or **New Management Profile**.
5. Enter a **Name** for the profile.
6. For **Permitted Services**, select services, such as **Ping**, and click **OK**.

STEP 11 | Commit your changes.

STEP 12 | Cable the interface.

Attach straight through cables from interfaces you configured to the corresponding switch or router on each network segment.

STEP 13 | Verify that the interface is active.

From the web interface, select **Network > Interfaces** and verify that icon in the Link State column is green. You can also monitor link state from the **Interfaces** widget on the **Dashboard**.

STEP 14 | Configure static routes and/or a dynamic routing protocol so that the virtual router or logical router can route traffic.

STEP 15 | Configure a default route.

[Configure a Static Route](#) or [Create a Static Route](#) for a logical router and set it as the default.

Manage IPv6 Hosts Using NDP

This topic describes how you can use NDP to provision IPv6 hosts; therefore, you don't need a separate DHCPv6 server for that purpose. It also explains how to use NDP to monitor IPv6 addresses, allowing you to quickly track the IPv6 address and MAC address of a device and the associated user who has violated a security rule.

- [IPv6 Router Advertisements for DNS Configuration](#)
- [Configure RDNS Servers and DNS Search List for IPv6 Router Advertisements](#)
- [NDP Monitoring](#)
- [Enable NDP Monitoring](#)

IPv6 Router Advertisements for DNS Configuration

The firewall implementation of [Neighbor Discovery](#) (ND) is enhanced so that you can provision IPv6 hosts with the Recursive DNS Server (RDNS) Option and DNS Search List (DNSSL) Option per [RFC 6106](#), [IPv6 Router Advertisement Options for DNS Configuration](#). When you [Configure Layer 3 Interfaces](#), you configure these DNS options on the firewall so the firewall can provision your IPv6 hosts; therefore you don't need a separate DHCPv6 server to provision the hosts. The firewall sends IPv6 Router Advertisements (RAs) containing these options to IPv6 hosts as part of their DNS configuration to fully provision them to reach internet services. Thus, your IPv6 hosts are configured with:

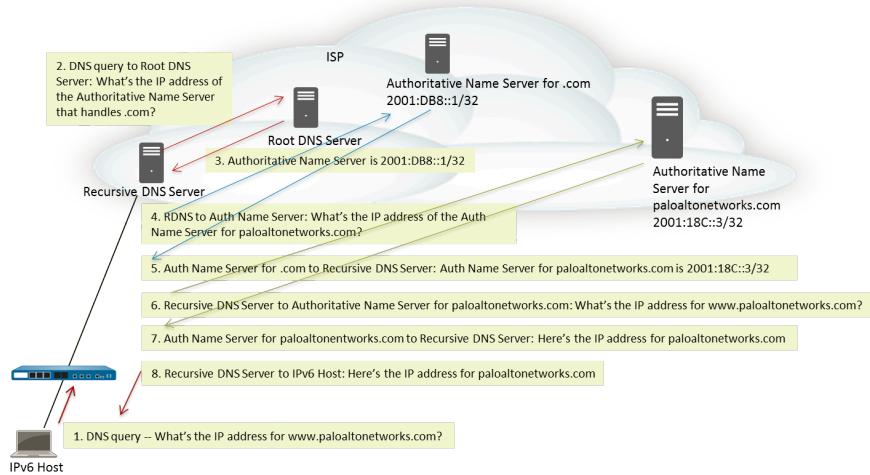
- The addresses of RDNS servers that can resolve DNS queries.
- A list of domain names (suffixes) that the DNS client appends (one at a time) to an unqualified domain name before entering the domain name into a DNS query.

IPv6 Router Advertisement for DNS configuration is supported for Ethernet interfaces, subinterfaces, Aggregated Ethernet interfaces, and Layer 3 VLAN interfaces on all PAN-OS platforms.



The capability of the firewall to send IPv6 RAs for DNS configuration allows the firewall to perform a role similar to DHCP, and is unrelated to the firewall being a DNS proxy, DNS client or DNS server.

After you configure the firewall with the addresses of RDNS servers, the firewall provisions an IPv6 host (the DNS client) with those addresses. The IPv6 host uses one or more of those addresses to reach an RDNS server. Recursive DNS refers to a series of DNS requests by an RDNS Server, as shown with three pairs of queries and responses in the following figure. For example, when a user tries to access www.paloaltonetworks.com, the local browser sees that it does not have the IP address for that domain name in its cache, nor does the client's operating system have it. The client's operating system launches a DNS query to a Recursive DNS Server belonging to the local ISP.



An IPv6 Router Advertisement can contain multiple DNS Recursive Server Address options, each with the same or different lifetimes. A single DNS Recursive DNS Server Address option can contain multiple Recursive DNS Server addresses as long as the addresses have the same lifetime.

A DNS Search List is a list of domain names (suffixes) that the firewall advertises to a DNS client. The firewall thus provisions the DNS client to use the suffixes in its unqualified DNS queries. The DNS client appends the suffixes, one at a time, to an unqualified domain name before it enters the name into a DNS query, thereby using a fully qualified domain name (FQDN) in the DNS query. For example, if a user (of the DNS client being configured) tries to submit a DNS query for the name "quality" without a suffix, the router appends a period and the first DNS suffix from the DNS Search List to the name and transmits a DNS query. If the first DNS suffix on the list is "company.com", the resulting DNS query from the router is for the FQDN "quality.company.com".

If the DNS query fails, the client appends the second DNS suffix from the list to the unqualified name and transmits a new DNS query. The client uses the DNS suffixes in order until a DNS lookup succeeds (ignoring the remaining suffixes) or the router has tried all of the suffixes on the list.

You configure the firewall with the suffixes that you want to provide to the DNS client router in an ND DNSSL option; the DNS client receiving the DNS Search List option is provisioned to use the suffixes in its unqualified DNS queries.

To specify RDNS Servers and a DNS Search List, [Configure RDNS Servers and DNS Search List for IPv6 Router Advertisements](#).

Configure RDNS Servers and DNS Search List for IPv6 Router Advertisements

Perform this task to configure [IPv6 Router Advertisements for DNS Configuration](#) of IPv6 hosts.

STEP 1 | Enable the firewall to send IPv6 Router Advertisements from an interface.

1. Select **Network > Interfaces and Ethernet or VLAN**.
2. Select the interface to configure.
3. On the **IPv6** tab, select **Enable IPv6 on the interface**.
4. On the **Router Advertisement** tab, select **Enable Router Advertisement**.
5. Click **OK**.

STEP 2 | Specify the Recursive DNS Server addresses and DNS Search List the firewall will advertise in ND Router Advertisements from this interface.

The RDNS servers and DNS Search List are part of the DNS configuration for the DNS client so that the client can resolve IPv6 DNS requests.

1. Select **Network > Interfaces and Ethernet or VLAN**.
2. Select the interface you are configuring.
3. Select **IPv6 > DNS Support**.
4. **Include DNS information in Router Advertisement** to enable the firewall to send IPv6 DNS information.
5. For **DNS Server**, Add the IPv6 address of a Recursive DNS Server. Add up to eight Recursive DNS servers. The firewall sends server addresses in an ICMPv6 Router Advertisement in order from top to bottom.
6. Specify the **Lifetime** in seconds, which is the maximum length of time the client can use the specific RDNS Server to resolve domain names.
 - The **Lifetime** range is any value equal to or between the **Max Interval** (that you configured on the **Router Advertisement** tab) and two times that **Max Interval**. For example, if your Max Interval is 600 seconds, the Lifetime range is 600-1,200 seconds.
 - The default **Lifetime** is 1,200 seconds.
7. For **DNS Suffix**, Add a **DNS Suffix** (domain name of a maximum of 255 bytes). Add up to eight DNS suffixes. The firewall sends suffixes in an ICMPv6 Router Advertisement in order from top to bottom.
8. Specify the **Lifetime** in seconds, which is the maximum length of time the client can use the suffix. The Lifetime has the same range and default value as the **Server**.
9. Click **OK**.

STEP 3 | Commit your changes.

Click **Commit**.

NDP Monitoring

Neighbor Discovery Protocol (NDP) for IPv6 ([RFC 4861](#)) performs functions similar to ARP functions for IPv4. The firewall by default runs NDP, which uses ICMPv6 packets to discover and track the link-layer addresses and status of neighbors on connected links.

[Enable NDP Monitoring](#) so you can view the IPv6 addresses of devices on the link local network, their MAC address, associated username from User-ID (if the user of that device used the directory service to log in), reachability Status of the address, and Last Reported date and time the NDP monitor received a Router Advertisement from this IPv6 address. The username is on a best-case basis; there can be many IPv6 devices on a network with no username, such as printers, fax machines, servers, etc.

If you want to quickly track a device and user who has violated a security rule, it is very useful to have the IPv6 address, MAC address and username displayed all in one place. You need the MAC address that corresponds to the IPv6 address in order to trace the MAC address back to a physical switch or Access Point.



NDP monitoring is not guaranteed to discover all devices because there could be other networking devices between the firewall and the client that filter out NDP or Duplicate Address Detection (DAD) messages. The firewall can monitor only the devices that it learns about on the interface.

NDP monitoring also monitors Duplicate Address Detection (DAD) packets from clients and neighbors. You can also monitor IPv6 ND logs to make troubleshooting easier.

NDP monitoring is supported for Ethernet interfaces, subinterfaces, Aggregated Ethernet interfaces, and VLAN interfaces on all PAN-OS models.

Enable NDP Monitoring

Perform this task to enable [NDP Monitoring](#) for an interface.

STEP 1 | Enable NDP monitoring.

1. Select **Network > Interfaces** and **Ethernet** or **VLAN**.
2. Select the interface you are configuring.
3. Select **IPv6**.
4. Select **Address Resolution**.
5. Select **Enable NDP Monitoring**.



After you enable or disable NDP monitoring, you must **Commit** before NDP monitoring can start or stop.

6. Click **OK**.

STEP 2 | Commit your changes.

Click **Commit**.

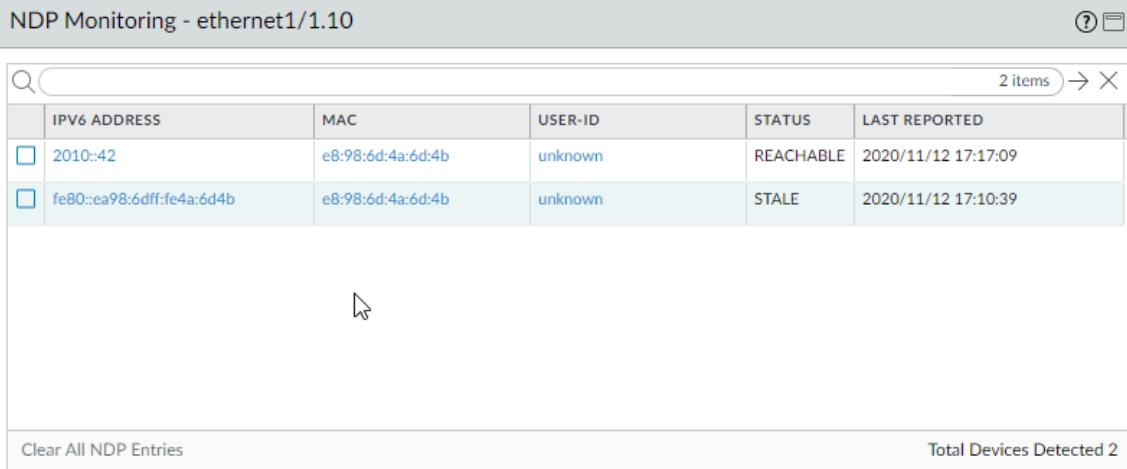
STEP 3 | Monitor NDP and DAD packets from clients and neighbors.

1. Select **Network > Interfaces** and **Ethernet or VLAN**.
2. For the interface where you enabled NDP monitoring, in the Features column, hover over the NDP Monitoring  icon:

The NDP Monitoring summary for the interface displays the list of **IPv6 Prefixes** that this interface will send in the Router Advertisement (RA) if RA is enabled (they are the IPv6 prefixes of the interface itself).

The summary also indicates whether DAD, Router Advertisement, and DNS Support are enabled; IP addresses of any Recursive DNS Servers configured; and any DNS suffixes configured on the DNS Search List.

3. Click on the NDP Monitoring icon to display detailed information.



The screenshot shows a table titled "NDP Monitoring - ethernet1/1.10". The table has columns: IPV6 ADDRESS, MAC, USER-ID, STATUS, and LAST REPORTED. There are two entries:

IPV6 ADDRESS	MAC	USER-ID	STATUS	LAST REPORTED
2010::42	e8:98:6d:4a:6d:4b	unknown	REACHABLE	2020/11/12 17:17:09
fe80::ea98:6dff:fe4a:6d4b	e8:98:6d:4a:6d:4b	unknown	STALE	2020/11/12 17:10:39

At the bottom left is a "Clear All NDP Entries" button. At the bottom right is a "Total Devices Detected 2" label and a "Close" button.

Each row of the detailed NDP Monitoring table for the interface displays the IPv6 address of a neighbor the firewall has discovered, the corresponding MAC address, corresponding User ID (on a best-case basis), reachability Status of the address, and Last Reported date and time this NDP Monitor received an RA from this IP address. A User ID will not display for printers or other non-user-based hosts. If the status of the IP address is Stale, the neighbor is not known to be reachable, per RFC 4861.

At the bottom right is the count of **Total Devices Detected** on the link local network.

- Enter an IPv6 address in the filter field to search for an address to display.
- Select the check boxes to display or not display IPv6 addresses.
- Click the numbers, the right or left arrow, or the vertical scroll bar to advance through many entries.
- Click **Clear All NDP Entries** to clear the entire table.

STEP 4 | Monitor ND logs for reporting purposes.

1. Select **Monitor > Logs > System**.
2. In the Type column, view **ipv6nd** logs and corresponding descriptions.

For example, `inconsistent router advertisement received` indicates that the firewall received an RA different from the RA that it is going to send out.

Configure an Aggregate Interface Group

An aggregate interface group uses IEEE 802.1AX link aggregation to combine multiple Ethernet interfaces into a single virtual interface that connects the firewall to another network device or firewall. An aggregate group increases the bandwidth between peers by load balancing traffic across the combined interfaces. It also provides redundancy; when one interface fails, the remaining interfaces continue supporting traffic.

By default, interface failure detection is automatic only at the physical layer between directly connected peers. However, if you enable Link Aggregation Control Protocol (LACP), failure detection is automatic at the physical and data link layers regardless of whether the peers are directly connected. LACP also enables automatic failover to standby interfaces if you configured hot spares. All Palo Alto Networks[®] firewalls except VM-Series models support aggregate groups. The [Product Selection tool](#) indicates the number of aggregate groups each firewall supports. Each aggregate group can have up to eight interfaces.



PAN-OS[®] firewall models support a maximum of 16,000 IP addresses assigned to physical or virtual Layer 3 interfaces; this maximum includes both IPv4 and IPv6 addresses.

QoS is supported on only the first eight aggregate groups.

Before configuring an aggregate group, you must configure its interfaces. Among the interfaces assigned to any particular aggregate group, the hardware media can differ (for example, you can mix fiber optic and copper), but the bandwidth and interface type must be the same. The bandwidth and interface type options are:

- **Bandwidth**—1Gbps, 10Gbps, 25Gbps, 40Gbps, or 100Gbps.
- **Interface type**—HA3, virtual wire, Layer 2, or Layer 3.



This procedure describes configuration steps only for the Palo Alto Networks firewall. You must also configure the aggregate group on the peer device. Refer to the documentation of that device for instructions.

STEP 1 | Configure the general interface group parameters.

1. Select **Network > Interfaces > Ethernet** and **Add Aggregate Group**.
2. In the field adjacent to the read-only **Interface Name**, enter a number to identify the aggregate group. The range is 1 to the maximum number of aggregate interface groups supported by the firewall.
3. For the **Interface Type**, select **HA**, **Virtual Wire**, **Layer2**, or **Layer3**.
4. Configure the remaining parameters for the **Interface Type** you selected.

STEP 2 | Configure the LACP settings.

Perform this step only if you want to enable LACP for the aggregate group.



You cannot enable LACP for virtual wire interfaces.

1. Select the **LACP** tab and **Enable LACP**.
2. Set the **Mode** for LACP status queries to **Passive** (the firewall just responds—the default) or **Active** (the firewall queries peer devices).
 As a best practice, set one LACP peer to active and the other to passive. LACP cannot function if both peers are passive. The firewall cannot detect the mode of its peer device.
3. Set the **Transmission Rate** for LACP query and response exchanges to **Slow** (every 30 seconds—the default) or **Fast** (every second). Base your selection on how much LACP processing your network supports and how quickly LACP peers must detect and resolve interface failures.
4. Select **Fast Failover** if you want to enable failover to a standby interface in less than one second. By default, the option is disabled and the firewall uses the IEEE 802.1ax standard for failover processing, which takes at least three seconds.
 *As a best practice, use **Fast Failover** in deployments where you might lose critical data during the standard failover interval.*
5. Enter the **Max Ports** (number of interfaces) that are active (1 to 8) in the aggregate group. If the number of interfaces you assign to the group exceeds the **Max Ports**, the remaining interfaces will be in standby mode. The firewall uses the **LACP Port Priority** of each interface you assign (Step 3) to determine which interfaces are initially active and to determine the order in which standby interfaces become active upon failover. If the LACP peers have non-matching port priority values, the values of the peer with the lower **System Priority** number (default is 32,768; range is 1 to 65,535) will override the other peer.
6. (**Optional**) For active/passive firewalls only, select **Enable in HA Passive State** if you want to enable LACP pre-negotiation for the passive firewall. LACP pre-negotiation enables quicker failover to the passive firewall (for details, see [LACP and LLDP Pre-Negotiation for Active/Passive HA](#)).
 *If you select this option, you cannot select **Same System MAC Address for Active-Passive HA**; pre-negotiation requires unique interface MAC addresses on each HA firewall.*
7. (**Optional**) For active/passive firewalls only, select **Same System MAC Address for Active-Passive HA** and specify a single **MAC Address** for both HA firewalls. This option minimizes failover latency if the LACP peers are virtualized (appearing to the network as a single device). By default, the option is disabled: each firewall in an HA pair has a unique MAC address.
 If the LACP peers are not virtualized, use unique MAC addresses to minimize failover latency.

STEP 3 | Click **OK**.

STEP 4 | Assign interfaces to the aggregate group.

Perform the following steps for each interface (1–8) that will be a member of the aggregate group.

1. Select **Network > Interfaces > Ethernet** and click the interface name to edit it.
2. Set the **Interface Type** to **Aggregate Ethernet**.
3. Select the **Aggregate Group** you just defined.
4. Select the **Link Speed**, **Link Duplex**, and **Link State**.



As a best practice, set the same link speed and duplex values for every interface in the group. For non-matching values, the firewall defaults to the higher speed and full duplex.

5. (**Optional**) Enter an **LACP Port Priority** (default is 32,768; range is 1 to 65,535) if you enabled LACP for the aggregate group. If the number of interfaces you assign exceeds the **Max Ports** value of the group, the port priorities determine which interfaces are active or standby. The interfaces with the lower numeric values (higher priorities) will be active.
6. Click **OK**.

STEP 5 | If the firewalls have an active/active configuration and you are aggregating HA3 interfaces, enable packet forwarding for the aggregate group.

1. Select **Device > High Availability > Active/Active Config** and edit the Packet Forwarding section.
2. Select the aggregate group you configured for the **HA3 Interface** and click **OK**.

STEP 6 | Commit your changes.

STEP 7 | Verify the aggregate group status.

1. Select **Network > Interfaces > Ethernet**.
2. Verify that the Link State column displays a green icon for the aggregate group, indicating that all member interfaces are up. If the icon is yellow, at least one member is down but not all. If the icon is red, all members are down.
3. If you configured LACP, verify that the Features column displays the LACP enabled icon for the aggregate group.

STEP 8 | (**PA-7050 and PA-7080 firewalls only**) If you have an aggregate interface group that has interfaces located on different line cards, it is a best practice to enable the firewall so that it can handle fragmented IP packets it receives on multiple interfaces of the AE group that are

spread over multiple cards. To do so, use the following CLI operational command with the **hash** keyword. (The other two keywords are also shown for completeness.)

1. [Access the CLI](#).
2. Use the following operational CLI command: **set ae-frag redistribution-policy <self|fixed sXdpX | hash>**
 - **self**—(default) This keyword is for legacy behavior; it does not enable the firewall to handle fragmented packets received on multiple interfaces of an AE interface group.
 - **fixed s<slot-number>dp<dataplane-cpu-number>**—Replace the *slot-number* variable and replace the *data-plane-cpu-number* variable with the dataplane number of the dataplane that will handle all IP fragments received by all members of all AE interfaces. The **fixed** keyword is intended mainly for troubleshooting purposes and shouldn't be used in production.
 - **hash**—Use to enable the firewall to handle fragmented packets it receives on multiple interfaces of an AE interface group that are located on more than one line card.

Configure Bonjour Reflector for Network Segmentation

Apple Bonjour (also known as zero-configuration networking) enables automatic discovery of devices and services on a local network. For example, Bonjour allows you to connect to a printer without manually configuring the printer's IP address. To translate names to addresses on a local network, Bonjour uses Multicast DNS (mDNS). Bonjour uses a private multicast range for its traffic, which does not allow traffic routing, preventing use in an environment that uses network segmentation for security or administrative purposes (for example, where servers and clients are in different subnets).

To support Apple Bonjour in network environments that use segmentation to route traffic, you can forward Bonjour IPv4 traffic between [Layer 3 Interfaces](#) (L3) Ethernet or [Aggregate Ethernet](#) (AE) interfaces or subinterfaces that you specify. The Bonjour Reflector option allows you to forward multicast Bonjour advertisements and queries to L3 Ethernet and AE interfaces or subinterfaces, ensuring user access to services and device discoverability regardless of Time To Live (TTL) values or hop limitations.



Bonjour traffic forwarding is supported for the PA-220, PA-400, PA-800, and PA-3200 series.

When you enable this option, the firewall redirects Bonjour traffic to the L3 and AE interfaces and subinterfaces where you enable this option. You must enable this option on all supported interfaces that you want to manage Bonjour traffic; for example, if you want a specific L3 interface to forward Bonjour traffic to an AE interface, you must enable this option on both interfaces. You can enable this option on up to 16 interfaces.



To prevent loops, the firewall modifies the source MAC address to the firewall's egress interface MAC address. To help prevent flooding attacks, if the firewall receives more than the number of packets per second specified in the following table, the firewall drops the packets to protect the firewall and the network.

Series	Rate Limit (per second)
PA-220	100
PA-400	N/A
PA-800	200
PA-3200	500

STEP 1 | Select **Network > Interfaces**.

STEP 2 | Select or **Add** an L3 ethernet or subinterface or AE interface.



*If you add a subinterface, it must use a **Tag** other than 0.*

Configure Interfaces

STEP 3 | Select IPv4 then select the **Enable Bonjour Reflector** option.

Ethernet Interface

Interface Name: ethernet1/3

Comment:

Interface Type: Layer3

Netflow Profile: None

Config | **IPv4** | IPv6 | SD-WAN | Advanced

Enable SD-WAN Enable Bonjour Reflector

Type: Static PPPoE DHCP Client

<input type="checkbox"/>	IP
<input type="checkbox"/>	

Add **Delete** **Move Up** **Move Down**

IP address/netmask. Ex. 192.168.2.254/24

OK **Cancel**

STEP 4 | Click OK.

STEP 5 | Repeat steps 1–4 for all L3 or AE interfaces and subinterfaces where you want to forward Bonjour traffic.

 You can enable this option on up to 16 different interfaces or subinterfaces.

STEP 6 | Commit your changes.

STEP 7 | Confirm that the **Features** column for the interface or interfaces where you enable the Bonjour Reflector option displays Bonjour Reflector:yes ().

STEP 8 | Use the `show bonjour interface` CLI command to display all interfaces where the firewall forwards Bonjour traffic and a list of counters. rx represents the total number of Bonjour packets the interface receives, tx represents the total number of Bonjour packets the interface transmits, and drop represents the number of packets the interface drops.

```
admin> show bonjour interface
```

name	rx	tx	drop
------	----	----	------

Configure Interfaces

ethernet1/4	1	1	0
ethernet1/7	0	0	0
ethernet1/7.10	0	0	0
ethernet1/7.20	4	4	0
ae15	0	0	0
ae16	0	0	0
ae16.30	0	2	0
ae16.40	0	0	0

Use Interface Management Profiles to Restrict Access

An Interface Management profile protects the firewall from unauthorized access by defining the protocols, services, and IP addresses that a firewall interface permits for management traffic. For example, you might want to prevent users from accessing the firewall web interface over the ethernet1/1 interface but allow that interface to receive SNMP queries from your network monitoring system. In this case, you would enable SNMP and disable HTTP/HTTPS in an Interface Management profile and assign the profile to ethernet1/1.

You can assign an Interface Management profile to Layer 3 Ethernet interfaces (including subinterfaces) and to logical interfaces (aggregate group, VLAN, loopback, and tunnel interfaces). If you do not assign an Interface Management profile to an interface, it denies access for all IP addresses, protocols, and services by default.



The management (MGT) interface does not require an Interface Management profile. You restrict protocols, services, and IP addresses for the MGT interface when you perform initial configuration of the firewall. In case the MGT interface goes down, allowing management access over another interface enables you to continue managing the firewall.



When enabling access to a firewall interface using an Interface Management profile, do not enable management access (HTTP, HTTPS, SSH, or Telnet) from the internet or from other untrusted zones inside your enterprise security boundary, and never enable HTTP or Telnet access because those protocols transmit in cleartext. Follow the [Best Practices for Securing Administrative Access](#) to ensure that you are properly securing management access to your firewall.

STEP 1 | Configure the Interface Management profile.

1. Select **Network > Network Profiles > Interface Mgmt** and click **Add**.
2. Select the protocols that the interface permits for management traffic: **Ping, Telnet, SSH, HTTP, HTTP OCSP, HTTPS, or SNMP**.



*Don't enable **HTTP** or **Telnet** because those protocols transmit in cleartext and therefore aren't secure.*

3. Select the services that the interface permits for management traffic:
 - **Response Pages**—Use to enable response pages for:
 - **Captive Portal**—To serve Captive Portal response pages, the firewall leaves ports open on Layer 3 interfaces: 6081 for Captive Portal in transparent mode and 6082

for Captive Portal in redirect mode. For details, see [Authentication Policy and Authentication Portal](#).

- **URL Admin Override**—For details, see [Allow Password Access to Certain Sites](#).
 - **User-ID**—Use to [Redistribute Data and Authentication Timestamps](#).
 - **User-ID Syslog Listener-SSL or User-ID Syslog Listener-UDP**—Use to [Configure User-ID to Monitor Syslog Senders for User Mapping](#) over SSL or UDP.
4. (**Optional**) Add the Permitted IP Addresses that can access the interface. If you don't add entries to the list, the interface has no IP address restrictions.
 5. Click **OK**.

STEP 2 | Assign the Interface Management profile to an interface.

1. Select **Network > Interfaces**, select the type of interface (**Ethernet**, **VLAN**, **Loopback**, or **Tunnel**), and select the interface.
2. Select **Advanced > Other info** and select the **Interface Management Profile** you just added.
3. Click **OK** and **Commit**.

Virtual Routers

Learn about how a virtual router on the firewall participates in Layer 3 routing and configure a virtual router.

- [Virtual Router Overview](#)
- [Configure Virtual Routers](#)

Virtual Router Overview

The firewall uses virtual routers to obtain Layer 3 routes to other subnets by you manually defining static routes or through participation in one or more Layer 3 routing protocols (dynamic routes). The routes that the firewall obtains through these methods populate the IP routing information base (RIB) on the firewall. When a packet is destined for a different subnet than the one it arrived on, the virtual router obtains the best route from the RIB, places it in the forwarding information base (FIB), and forwards the packet to the next hop router defined in the FIB. The firewall uses Ethernet switching to reach other devices on the same IP subnet. (An exception to one best route going in the FIB occurs if you are using [ECMP](#), in which case all equal-cost routes go in the FIB.)

The Ethernet, VLAN, and tunnel interfaces defined on the firewall receive and forward Layer 3 packets. The destination zone is derived from the outgoing interface based on the forwarding criteria, and the firewall consults policy rules to identify the security policies that it applies to each packet. In addition to routing to other network devices, virtual routers can route to other virtual routers within the same firewall if a next hop is specified to point to another virtual router.

You can [configure Layer 3 interfaces on a virtual router](#) to participate with dynamic routing protocols (BGP, OSPF, OSPFv3, or RIP) as well as add static routes. You can also create multiple virtual routers, each maintaining a separate set of routes that aren't shared between virtual routers, enabling you to configure different routing behaviors for different interfaces.

You can configure dynamic routing from one virtual router to another by configuring a loopback interface in each virtual router, creating a static route between the two loopback interfaces, and then configuring a dynamic routing protocol to peer between these two interfaces. The firewall supports only one hop between virtual routers. For example, with virtual routers A, B, and C, a route cannot go from A to B to C; it would have to go from A to C.

Each Layer 3 Ethernet, loopback, VLAN, and tunnel interface defined on the firewall must be associated with a virtual router. While each interface can belong to only one virtual router, you can configure multiple routing protocols and static routes for a virtual router. Regardless of the static routes and dynamic routing protocols you configure for a virtual router, one general configuration is required.

Configure Virtual Routers

Create a [virtual router](#) on the firewall to participate in Layer 3 routing.

STEP 1 | Gather the required information from your network administrator.

- Interfaces on the firewall that you want to perform routing.
- Administrative distances for static, OSPF internal, OSPF external, IBGP, EBGP and RIP.

STEP 2 | Create a virtual router and apply interfaces to it.

The firewall comes with a virtual router named **default**. You can edit the **default** virtual router or add a new virtual router.

1. Select **Network > Virtual Routers**.
2. Select a virtual router (the one named **default** or a different virtual router) or **Add** the **Name** of a new virtual router.
3. Select **Router Settings > General**.
4. Click **Add** in the **Interfaces** box and select an already defined interface.
Repeat this step for all interfaces you want to add to the virtual router.
5. Click **OK**.

STEP 3 | Set Administrative Distances for static and dynamic routing.

Set Administrative Distances for types of routes as required for your network. When the virtual router has two or more different routes to the same destination, it uses administrative distance to choose the best path from different routing protocols and static routes, by preferring a lower distance.

- **Static**—Range is 10 to 240; default is 10.
- **OSPF Internal**—Range is 10 to 240; default is 30.
- **OSPF External**—Range is 10 to 240; default is 110.
- **IBGP**—Range is 10 to 240; default is 200.
- **EBGP**—Range is 10 to 240; default is 20.
- **RIP**—Range is 10 to 240; default is 120.



See [ECMP](#) if you want to leverage having multiple equal-cost paths for forwarding.

STEP 4 | Commit virtual router general settings.

Click **OK** and **Commit**.

STEP 5 | Configure Ethernet, VLAN, loopback, and tunnel interfaces as needed.

[Configure Layer 3 Interfaces](#).

Service Routes

Learn about how the firewall uses service routes to send requests to external services and configure service routes.

- [Service Routes Overview](#)
- [Configure Service Routes](#)

Service Routes Overview

The firewall uses the management (MGT) interface by default to access external services, such as DNS servers, external authentication servers, Palo Alto Networks® services such as software, URL updates, licenses and AutoFocus. An alternative to using the MGT interface is to configure a data port (a regular interface) to access these services. The path from the interface to the service on a server is known as a *service route*. The service packets exit the firewall on the port assigned for the external service and the server sends its response to the configured source interface and source IP address.

You can [Configure Service Routes](#) globally for the firewall or [customize service routes for a virtual system](#) on a firewall enabled for multiple virtual systems so that you have the flexibility to use interfaces associated with a virtual system. Any virtual system that does not have a service route configured for a particular service inherits the interface and IP address that are set globally for that service.

Configure Service Routes

The following procedure enables you to configure [service routes](#) to change the interface that the firewall uses to send requests to external services such as the Palo Alto Network cloud services or for log forwarding. For firewalls in a [high availability \(HA\)](#) configuration, the service route configuration is synchronized across the HA peers.

For firewalls in an [active/passive high availability \(HA\)](#), the service route you configured to leverage an external service or for log forwarding sees activity only on the active HA peer while the passive HA peer sees no activity if you configured an Ethernet interface as the **Source Interface**. For example, you configure a service route with Ethernet 1/3 as the source interface to forward logs to Strata Logging Service. In this scenario, all logs are forwarded from the active HA peer but no logs, including the system and configuration logs, are forwarded from the passive HA peer. However, if you configure the MGT interface as the service route **Source Interface**, activity occurs on both the active and passive HA peers.

STEP 1 | Customize service routes.

1. Select **Device > Setup > Services > Global** (omit Global on a firewall without multiple virtual system capability), and in the Services Features section, click **Service Route Configuration**.



2. Select **Customize** and do one of the following to create a service route:
 - For a predefined service:
 - Select **IPv4** or **IPv6** and click the link for the service for which you want customize the service route.
 -  To easily use the same source address for multiple services, select the checkbox for the services, click **Set Selected Routes**, and proceed to the next step.
 - To limit the list for Source Address, select a **Source Interface**; then select a **Source Address** (from that interface) as the service route. An Address Object can also be referenced as a Source Address if it is already configured on the selected interface. Selecting **Any** Source Interface makes all IP addresses on all interfaces available in the Source Address list from which you select an address. Selecting **Use default** causes the firewall to use the management interface for the service route, unless the packet destination IP address matches the configured Destination IP address, in which case the source IP address is set to the **Source Address** configured for the

Destination. Selecting **MGT** causes the firewall to use the MGT interface for the service route, regardless of any destination service route.



The Service Route Source Address does not inherit configuration changes from the referenced interface and vice versa. Modification of an Interface IP Address to a different IP address or Address Object will not update a corresponding Service Route Source Address. This may lead to commit failure and require you to update the Service Route(s) to a valid Source Address value.

- Click **OK** to save the setting.
- Repeat this step if you want to specify both an IPv4 and IPv6 address for a service.
- For a destination service route:
 - Select **Destination** and **Add a Destination** IP address. In this case, if a packet arrives with a destination IP address that matches this configured **Destination** address, then the source IP address of the packet will be set to the **Source Address** configured in the next step.
 - To limit the list for Source Address, select a **Source Interface**; then select a **Source Address** (from that interface) as the service route. Selecting **Any** Source Interface makes all IP addresses on all interfaces available in the Source Address list from which you select an address. Selecting **MGT** causes the firewall to use the MGT interface for the service route.
 - Click **OK** to save the setting.
- 3. Repeat the prior steps for each service route you want to customize.
- 4. Click **OK** to save the service route configuration.

STEP 2 | Commit.

Static Routes

Static routes are typically used in conjunction with dynamic routing protocols. You might configure a static route for a location that a dynamic routing protocol can't reach. Static routes require manual configuration on every router in the network, rather than the firewall entering dynamic routes in its route tables; even though static routes require that configuration on all routers, they may be desirable in small networks rather than configuring a routing protocol.

- [Static Route Overview](#)
- [Static Route Removal Based on Path Monitoring](#)
- [Configure a Static Route](#)
- [Configure Path Monitoring for a Static Route](#)

Static Route Overview

If you decide that you want specific Layer 3 traffic to take a certain route without participating in IP routing protocols, you can [Configure a Static Route](#) using IPv4 and IPv6 routes.

A default route is a specific static route. If you don't use dynamic routing to obtain a default route for your virtual router, you must configure a static default route. When the virtual router has an incoming packet and finds no match for the packet's destination in its route table, the virtual router sends the packet to the default route. The default IPv4 route is 0.0.0.0/0; the default IPv6 route is ::/0. You can configure both an IPv4 and IPv6 default route.

Static routes themselves don't change or adjust to changes in network environments, so traffic typically isn't rerouted if a failure occurs along the route to a statically defined endpoint. However, you have options to back up static routes in the event of a problem:

- You can configure a static route with a Bidirectional Forwarding Detection ([BFD](#)) profile so that if a BFD session between the firewall and the BFD peer fails, the firewall removes the failed static route from the RIB and FIB tables and uses an alternative route with a lower priority.
- You can [Configure Path Monitoring for a Static Route](#) so that the firewall can use an alternative route.

By default, static routes have an administrative distance of 10. When the firewall has two or more routes to the same destination, it uses the route with the lowest administrative distance. By increasing the administrative distance of a static route to a value higher than a dynamic route, you can use the static route as a backup route if the dynamic route is unavailable.

While you're configuring a static route, you can specify whether the firewall installs an IPv4 static route in the unicast or multicast route table (RIB), or both tables, or doesn't install the route at all. For example, you could install an IPv4 static route in the multicast route table only, because you want only multicast traffic to use that route. This option give you more control over which route the traffic takes. You can specify whether the firewall installs an IPv6 static route in the unicast route table or not.

Static Route Removal Based on Path Monitoring

When you [Configure Path Monitoring for a Static Route](#), the firewall uses path monitoring to detect when the path to one or more monitored destination has gone down. The firewall can then reroute traffic using alternative routes. The firewall uses path monitoring for static routes much like path monitoring for HA or policy-based forwarding (PBF), as follows:

- The firewall sends ICMP ping messages (heartbeat messages) to one or more monitored destinations that you determine are robust and reflect the availability of the static route.
- If pings to any or all of the monitored destinations fail, the firewall considers the static route down too and removes it from the Routing Information Base (RIB) and Forwarding Information Base (FIB). The RIB is the table of static routes the firewall is configured with and dynamic routes it has learned from routing protocols. The FIB is the forwarding table of routes the firewall uses for forwarding packets. The firewall selects an alternative static route to the same destination (based on the route with the lowest metric) from the RIB and places it in the FIB.
- The firewall continues to monitor the failed route. When the route comes back up, and (based on the **Any** or **All** failure condition) the path monitor returns to Up state, the preemptive hold timer begins. The path monitor must remain up for the duration of the hold timer; then the firewall considers the static route stable and reinstates it into the RIB. The firewall then compares metrics of routes to the same destination to decide which route goes in the FIB.

Path monitoring is a desirable mechanism to avoid silently discarding traffic for:

- A static or default route.
- A static or default route redistributed into a routing protocol.
- A static or default route when one peer does not support BFD. (The best practice is not to enable both BFD and path monitoring on a single interface.)
- A static or default route instead of using PBF path monitoring, which doesn't remove a failed static route from the RIB, FIB, or redistribution policy.

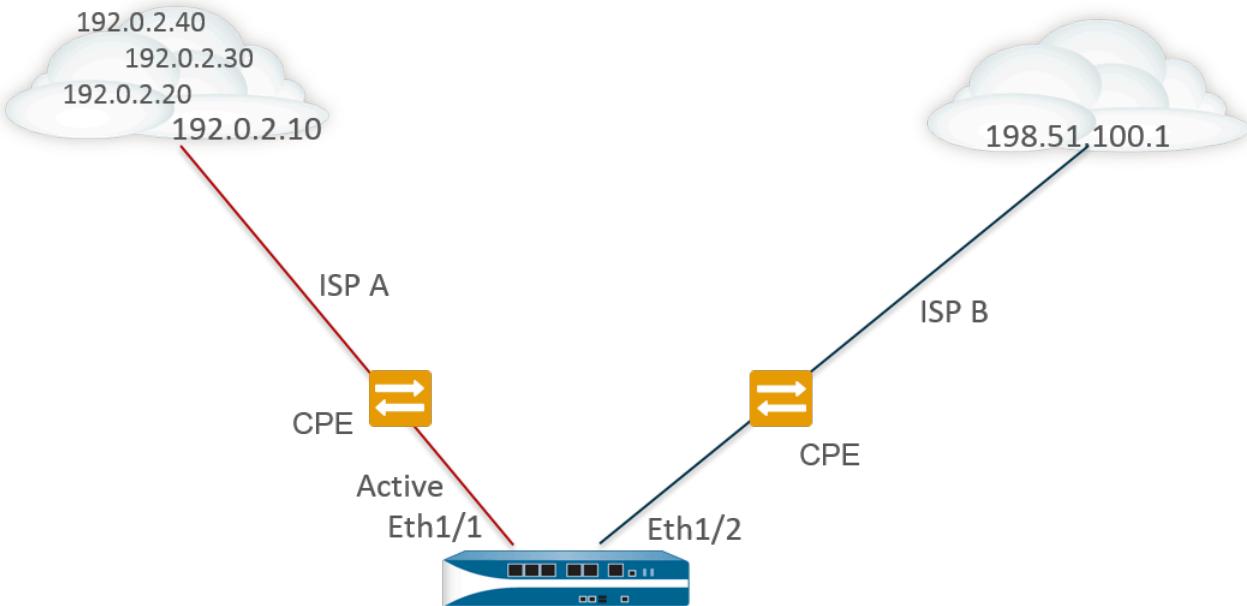


Path monitoring doesn't apply to static routes configured between virtual routers.

In the following figure, the firewall is connected to two ISPs for route redundancy to the internet. The primary default route 0.0.0.0 (metric 10) uses Next Hop 192.0.2.10; the secondary default route 0.0.0.0 (metric 50) uses Next Hop 198.51.100.1. The customer premises equipment (CPE) for ISP A keeps the primary physical link active, even after internet connectivity goes down. With the link artificially active, the firewall can't detect that the link is down and that it should replace the failed route with the secondary route in its RIB.

To avoid silently discarding traffic to a failed link, configure path monitoring of 192.0.2.20, 192.0.2.30, and 192.0.2.40 and if all (or any) of the paths to these destinations fail, the firewall presumes the path to Next Hop 192.0.2.10 is also down, removes the static route 0.0.0.0 (that uses Next Hop 192.0.2.10) from its RIB, and replaces it with the secondary route to the same destination 0.0.0.0 (that uses Next Hop 198.51.100.1), which also accesses the internet.

Static Routes



Route Table

Destination

Destination	Next Hop	Metric	Interface
0.0.0.0/0	192.0.2.10	10	ethernet1/1
0.0.0.0/0	198.51.100.1	50	ethernet1/2

When you [Configure a Static Route](#), one of the required fields is the Next Hop toward that destination. The type of next hop you configure determines the action the firewall takes during path monitoring, as follows:

If Next Hop Type in Static Route is:	Firewall Action for ICMP Ping
IP Address	The firewall uses the source IP address and egress interface of the static route as the source address and egress interface in the ICMP ping. It uses the configured Destination IP address of the monitored destination as the ping's destination address. It uses the static route's next hop address as the ping's next hop address.
Next VR	The firewall uses the source IP address of the static route as the source address in the ICMP ping. The egress interface is based on the lookup result from the next hop's virtual router. The configured Destination IP address of the monitored destination is the ping's destination address.
None	The firewall uses the destination IP address of the path monitor as the next hop and sends the ICMP ping to the interface specified in the static route.

When path monitoring for a static or default route fails, the firewall logs a critical event (path-monitor-failure). When the static or default route recovers, the firewall logs another critical event (path-monitor-recovery).

Firewalls synchronize path monitoring configurations for an active/passive HA deployment, but the firewall blocks egress ICMP ping packets on a passive HA peer because it is not actively processing traffic. The firewall doesn't synchronize path monitoring configurations for active/active HA deployments.

Configure a Static Route

Perform the following task to configure [Static Routes](#) or a default route for a virtual router on the firewall.

STEP 1 | Configure a static route.

1. Select **Network > Virtual Router** and select the virtual router you are configuring, such as **default**.
2. Select the **Static Routes** tab.
3. Select **IPv4** or **IPv6**, depending on the type of static route you want to configure.
4. **Add a Name** (a maximum of 63 characters) for the route. The name must start with an alphanumeric character and can contain a combination of alphanumeric characters, underscore (_), hyphen (-), dot (.), and space.
5. For **Destination**, enter the route and netmask (for example, 192.168.2.2/24 for an IPv4 address or 2001:db8:123:1::1/64 for an IPv6 address). If you're creating a default route, enter the default route (0.0.0.0/0 for an IPv4 address or ::/0 for an IPv6 address). Alternatively, you can create an address object of type IP Netmask.
6. (**Optional**) For **Interface**, specify the outgoing interface for packets to use to go to the next hop. Use this for stricter control over which interface the firewall uses rather than the interface in the route table for the next hop of this route.
7. For **Next Hop**, select one of the following:
 - **IP Address**—Enter the IP address (for example, 192.168.56.1 or 2001:db8:49e:1::1) when you want to route to a specific next hop. You must **Enable IPv6 on the interface** (when you [Configure Layer 3 Interfaces](#)) to use an IPv6 next hop address. If you're creating a default route, for **Next Hop** you must select **IP Address** and enter the IP address for your Internet gateway (for example, 192.168.56.1 or 2001:db8:49e:1::1).

Alternatively, you can create an address object of type IP Netmask. The address object must have a netmask of /32 for IPv4 or /128 for IPv6.



While configuring static routes for a virtual router on the firewall, you can enter an IP address for the Next Hop router. Palo Alto Networks firewall treats the **Next Hop IP address** as an address object. Therefore, if you configure the Next Hop IP address (**Network > Virtual Router > Static Routes**) value same as the configured Address object name (**Objects > Addresses**), then any modifications to the address object will reflect in the Next Hop IP address value also. That is, renaming the address object (**Objects > Addresses**) will also rename the Next Hop IP address.

- **Next VR**—Select this option and then select a virtual router if you want to route internally to a different virtual router on the firewall.
- **FQDN**—Enter an FQDN or select an address object that uses an FQDN, or create a new address object of type FQDN.



If you use an FQDN as a static route next hop, that FQDN must resolve to an IP address that belongs to the same subnet as the interface you configured for the static route; otherwise, the firewall rejects the resolution and the FQDN remains unresolved.



The firewall uses only one IP address (from each IPv4 or IPv6 family type) from the DNS resolution of the FQDN. If the DNS resolution returns more than one address, the firewall uses the preferred IP address that matches the IP family type (IPv4 or IPv6) configured for the next hop. The preferred IP address is the first address the DNS server returns in its initial response. The firewall retains this address as preferred as long as the address appears in subsequent responses, regardless of its order.

- **Discard**—Select to drop packets that are addressed to this destination.
 - **None**—Select if there is no next hop for the route. For example, a point-to-point connection does not require a next hop because there is only one way for packets to go.
8. Enter an **Admin Distance** for the route to override the default administrative distance set for static routes for this virtual router (range is 10 to 240; default is 10).
 9. Enter a **Metric** for the route (range is 1 to 65,535).

STEP 2 | Choose where to install the route.

Select the **Route Table** (the RIB) into which you want the firewall to install the static route:

- **Unicast**—Install the route in the unicast route table. Choose this option if you want the route used only for unicast traffic.
- **Multicast**—Install the route in the multicast route table (available for IPv4 routes only). Choose this option if you want the route used only for multicast traffic.
- **Both**—Install the route in the unicast and multicast route tables (available for IPv4 routes only). Choose this option if you want either unicast or multicast traffic to use the route.
- **No Install**—Do not install the route in either route table.

STEP 3 | **(Optional)** If your firewall model supports **BFD**, you can apply a **BFD Profile** to the static route so that if the static route fails, the firewall removes the route from the RIB and FIB and uses an alternative route. Default is **None**.

STEP 4 | Click **OK** twice.

STEP 5 | Commit the configuration.

Configure Path Monitoring for a Static Route

Use the following procedure to configure [Static Route Removal Based on Path Monitoring](#).

STEP 1 | Enable path monitoring for a static route.

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **Static Routes**, select **IPv4** or **IPv6**, and select the static route you want to monitor. You can monitor up to 128 static routes.
3. Select **Path Monitoring** to enable path monitoring for the route.

STEP 2 | Configure the monitored destination(s) for the static route.

1. Add a monitored destination by **Name**. You can add up to eight monitored destinations per static route.
2. Select **Enable** to monitor the destination.
3. For **Source IP**, select the IP address that the firewall uses in the ICMP ping to the monitored destination:
 - If the interface has multiple IP addresses, select one.
 - If you select an interface, the firewall uses the first IP address assigned to the interface by default.
 - If you select **DHCP (Use DHCP Client address)**, the firewall uses the address that DHCP assigned to the interface. To see the DHCP address, select **Network > Interfaces > Ethernet** and in the row for the Ethernet interface, click on **Dynamic DHCP Client**. The IP Address displays in the Dynamic IP Interface Status window.
4. For **Destination IP**, enter an IP address or address object to which the firewall will monitor the path. The monitored destination and static route destination must use the same address family (IPv4 or IPv6).



The destination IP address should belong to a reliable endpoint; you wouldn't want to base path monitoring on a device that itself is unstable or unreliable.

5. **(Optional)** Specify the **ICMP Ping Interval (sec)** in seconds to determine how frequently the firewall monitors the path (range is 1-60; default is 3).
6. **(Optional)** Specify the **ICMP Ping Count** of packets that don't return from the destination before the firewall considers the static route down and removes it from the RIB and FIB (range is 3-10; default is 5).
7. Click **OK**.

STEP 3 | Determine whether path monitoring for the static route is based on one or all monitored destinations, and set the preemptive hold time.

1. Select a **Failure Condition**, whether **Any** or **All** of the monitored destinations for the static route must be unreachable by ICMP for the firewall to remove the static route

from the RIB and FIB and add the static route that has the next lowest metric going to the same destination to the FIB.



Select **All** to avoid the possibility of any single monitored destination signaling a route failure when the destination is simply offline for maintenance, for example.

2. **(Optional)** Specify the **Preemptive Hold Time (min)**, which is the number of minutes a downed path monitor must remain in Up state before the firewall reinstalls the static route into the RIB. The path monitor evaluates all of its monitored destinations for the static route and comes up based on the **Any** or **All** failure condition. If a link goes down or flaps during the hold time, when the link comes back up, the path monitor can come back up; the timer restarts when the path monitor returns to Up state.

A **Preemptive Hold Time** of zero causes the firewall to reinstall the route into the RIB immediately upon the path monitor coming up. Range is 0-1,440; default is 2.

3. Click **OK**.

STEP 4 | Commit.

Click **Commit**.

STEP 5 | Verify path monitoring on static routes.

1. Select **Network > Virtual Routers** and in the row of the virtual router you are interested in, select **More Runtime Stats**.
2. From the **Routing** tab, select **Static Route Monitoring**.
3. For a static route (Destination), view whether Path Monitoring is Enabled or Disabled. The Status column indicates whether the route is Up, Down, or Disabled. Flags for the static route are: A—active, S—static, E—ECMP.
4. Select **Refresh** periodically to see the latest state of the path monitoring (health check).
5. Hover over the Status of a route to view the monitored IP addresses and results of the pings sent to the monitored destinations for that route. For example, 3/5 means that a ping interval of 3 seconds and a ping count of 5 consecutive missed pings (the firewall receives no ping in the last 15 seconds) indicates path monitoring detects a link failure. Based on the **Any** or **All** failure condition, if path monitoring is in failed state and the firewall receives a ping after 15 seconds, the path can be deemed up and the **Preemptive Hold Time** starts.

The State indicates the last monitored ping results: success or failed. Failed indicates that the series of ping packets (ping interval multiplied by ping count) was not successful. A single ping packet failure does not reflect a failed ping state.

STEP 6 | View the RIB and FIB to verify that the static route is removed.

1. Select **Network > Virtual Routers** and in the row of the virtual router you are interested in, select **More Runtime Stats**.
2. From the **Routing** tab, select **Route Table** (RIB) and then the **Forwarding Table** (FIB) to view each, respectively.
3. Select **Unicast** or **Multicast** to view the appropriate route table.
4. For **Display Address Family**, select **IPv4 and IPv6**, **IPv4 Only**, or **IPv6 Only**.
5. (**Optional**) In the filter field, enter the route you are searching for and select the arrow, or use the scroll bar to move through pages of routes.
6. See if the route is removed or present.
7. Select **Refresh** periodically to see the latest state of the path monitoring (health check).



To view the events logged for path monitoring, select **Monitor > Logs > System**.

View the entry for **path-monitor-failure**, which indicates path monitoring for a static route destination failed, so the route was removed. View the entry for **path-monitor-recovery**, which indicates path monitoring for the static route destination recovered, so the route was restored.

RIP

Consider whether RIP is an appropriate routing protocol for your network and if so, configure RIP.

- [RIP Overview](#)
- [Configure RIP](#)

RIP Overview

Routing Information Protocol (RIP) is an interior gateway protocol (IGP) that was designed for small IP networks. RIP relies on hop count to determine routes; the best routes have the fewest number of hops. RIP is based on UDP and uses port 520 for route updates. By limiting routes to a maximum of 15 hops, the protocol helps prevent the development of routing loops, but also limits the supported network size. Before you [configure RIP](#), consider that if more than 15 hops are required, traffic is not routed. RIP also can take longer to converge than OSPF and other routing protocols.

The firewall supports RIP v2.

Configure RIP

Perform the following procedure to configure **RIP**.

STEP 1 | Configure general **virtual router** settings.

STEP 2 | Configure general RIP configuration settings.

1. Select a virtual router (**Network > Virtual Routers**) and for the virtual router, select **RIP**.
2. Select **Enable** to enable the RIP protocol.
3. Select **Reject Default Route** if you do not want to learn any default routes through RIP. This is the recommended, default setting.

Clear **Reject Default Route** if you want to permit redistribution of default routes through RIP.

STEP 3 | Configure interfaces for RIP.

1. On the **Interfaces** tab, select an interface in the Interface configuration section.
2. Select an already defined interface.
3. Select **Enable**.
4. Select **Advertise Default Route** to advertise a default route to RIP peers with the specified metric value.
5. **(Optional)** Select a profile from the **Auth Profile** list.
6. Select normal, passive or send-only from the **Mode** list.
7. **(Optional)** To enable **BFD** for RIP globally for the virtual router, select a **BFD** profile.
8. Click **OK**.

STEP 4 | Configure RIP timers.

1. On the **Timers** tab, enter a value for **Interval Seconds (sec)**. This setting defines the length of the following RIP timer intervals in seconds (range is 1 to 60; default is 1).
2. Specify the **Update Intervals** to define the number of intervals between route update announcements (range is 1 to 3,600; default is 30).
3. Specify the **Expire Intervals** to define the number of intervals between the time that the route was last updated to its expiration (range is 1 to 3600; default is 120).
4. Specify the **Delete Intervals** to define the number of intervals between the time that the route expires to its deletion (range is 1 to 3,600; default is 180).

STEP 5 | **(Optional)** Configure Auth Profiles.

By default, the firewall does not use RIP authentication for the exchange between RIP neighbors. Optionally, you can configure RIP authentication between RIP neighbors by either

a simple password or MD5 authentication. MD5 authentication is recommended; it is more secure than a simple password.

Simple Password RIP authentication

1. Select **Auth Profiles** and **Add** a name for the authentication profile to authenticate RIP messages.
2. Select **Simple Password** as the **Password Type**.
3. Enter a simple password and then confirm.

MD5 RIP authentication

1. Select **Auth Profiles** and **Add** a name for the authentication profile to authenticate RIP messages.
2. Select **MD5** as the **Password Type**.
3. **Add** one or more password entries, including:
 - Key-ID (range is 0 to 255)
 - Key
4. (**Optional**) Select **Preferred** status.
5. Click **OK** to specify the key to be used to authenticate outgoing message.
6. Click **OK** again in the Virtual Router - RIP Auth Profile dialog box.

STEP 6 | Commit your changes.

OSPF

Open Shortest Path First (OSPF) is an interior gateway protocol (IGP) that is most often used to dynamically manage network routes in large enterprise networks. It determines routes dynamically by obtaining information from other routers and advertising routes to other routers by way of Link State Advertisements (LSAs). The information gathered from the LSAs is used to construct a topology map of the network. This topology map is shared across routers in the network and used to populate the IP routing table with available routes.

Changes in the network topology are detected dynamically and used to generate a new topology map within seconds. A shortest path tree is computed of each route. Metrics associated with each routing interface are used to calculate the best route. These can include distance, network throughput, link availability etc. Additionally, these metrics can be configured statically to direct the outcome of the OSPF topology map.

The Palo Alto Networks® implementation of OSPF fully supports the following RFCs:

- [RFC 2328](#) (for IPv4)
- [RFC 5340](#) (for IPv6)

The following topics provide more information about the OSPF and procedures for configuring OSPF on the firewall:

- [OSPF Concepts](#)
- [Configure OSPF](#)
- [Configure OSPFv3](#)
- [Configure OSPF Graceful Restart](#)
- [Confirm OSPF Operation](#)

OSPF Concepts

OSPF determines routes dynamically by obtaining information from other routers and advertising routes to other routers by way of Link State Advertisements (LSAs). The router keeps information about the links between it and the destination and can make highly efficient routing decisions. A cost is assigned to each router interface, and the best routes are determined to be those with the lowest costs, when summed over all the encountered outbound router interfaces and the interface receiving the LSA.

Hierarchical techniques are used to limit the number of routes that must be advertised and the associated LSAs. Because OSPF dynamically processes a considerable amount of route information, it has greater processor and memory requirements than does RIP.

The following topics introduce the OSPF concepts you must understand in order to configure the firewall to participate in an OSPF network:

- [OSPFv3](#)
- [OSPF Neighbors](#)
- [OSPF Areas](#)
- [OSPF Router Types](#)

OSPFv3

OSPFv3 provides support for the OSPF routing protocol within an IPv6 network. As such, it provides support for IPv6 addresses and prefixes. It retains most of the structure and functions in OSPFv2 (for IPv4) with some minor changes. The following are some of the additions and changes to OSPFv3:

- **Support for multiple instances per link**—With OSPFv3, you can run multiple instances of the OSPF protocol over a single link. This is accomplished by assigning an OSPFv3 instance ID number. An interface that is assigned to an instance ID drops packets that contain a different ID.
- **Protocol Processing Per-link**—OSPFv3 operates per-link instead of per-IP-subnet as on OSPFv2.
- **Changes to Addressing**—IPv6 addresses are not present in OSPFv3 packets, except for LSA payloads within link state update packets. Neighboring routers are identified by the Router ID.
- **Authentication Changes**—OSPFv3 doesn't include any authentication capabilities. Configuring OSPFv3 on a firewall requires an authentication profile that specifies Encapsulating Security Payload (ESP) or IPv6 Authentication Header (AH). The re-keying procedure specified in RFC 4552 is not supported in this release.
- **Support for multiple instances per-link**—Each instance corresponds to an instance ID contained in the OSPFv3 packet header.
- **New LSA Types**—OSPFv3 supports two new LSA types: Link LSA and Intra Area Prefix LSA.

All additional changes are described in detail in RFC 5340.

OSPF Neighbors

Two OSPF-enabled routers connected by a common network and in the same OSPF area that form a relationship are OSPF neighbors. The connection between these routers can be through a common broadcast domain or by a point-to-point connection. This connection is made through the exchange of hello OSPF protocol packets. These neighbor relationships are used to exchange routing updates between routers.

OSPF Areas

OSPF operates within a single autonomous system (AS). Networks within this single AS, however, can be divided into a number of areas. By default, Area 0 is created. Area 0 can either function alone or act as the OSPF backbone for a larger number of areas. Each OSPF area is named using a 32-bit identifier which in most cases is written in the same dotted-decimal notation as an IP4 address. For example, Area 0 is usually written as 0.0.0.0.

The topology of an area is maintained in its own link state database and is hidden from other areas, which reduces the amount of traffic routing required by OSPF. The topology is then shared in a summarized form between areas by a connecting router.

OSPF Area Type	Description
Backbone Area	The backbone area (Area 0) is the core of an OSPF network. All other areas are connected to it and all traffic between areas must traverse it. All routing between areas is distributed through the backbone area. While all other OSPF areas must connect to the backbone area, this connection doesn't need to be direct and can be made through a virtual link.
Normal OSPF Area	In a normal OSPF area there are no restrictions; the area can carry all types of routes.
Stub OSPF Area	A stub area does not receive routes from other autonomous systems. Routing from the stub area is performed through the default route to the backbone area.
NSSA Area	The Not So Stubby Area (NSSA) is a type of stub area that can import external routes, with some limited exceptions.

OSPF Router Types

Within an OSPF area, routers are divided into the following categories.

- **Internal Router**—A router with that has OSPF neighbor relationships only with devices in the same area.
- **Area Border Router (ABR)**—A router that has OSPF neighbor relationships with devices in multiple OSPF areas. ABRs gather topology information from their connected areas and distribute it to the backbone area.

- **Backbone Router**—A backbone router is a router that runs OSPF and has at least one interface connected to the OSPF backbone area. Since ABRs are always connected to the backbone, they are always classified as backbone routers.
- **Autonomous System Boundary Router (ASBR)**—An ASBR is a router that attaches to more than one routing protocol and exchanges routing information between them.

Configure OSPF

After you understand [OSPF Concepts](#), perform the following procedure to configure OSPF.

STEP 1 | Configure general [virtual router](#) settings.

STEP 2 | Enable OSPF.

1. Select the **OSPF** tab.
2. Select **Enable** to enable the OSPF protocol.
3. Enter the **Router ID**.
4. Select **Reject Default Route** if you do not want to learn any default routes through OSPF. This is the recommended, default setting.

Clear **Reject Default Route** if you want to permit redistribution of default routes through OSPF.

STEP 3 | Configure Areas - Type for the OSPF protocol.

1. On the **Areas** tab, **Add** an **Area ID** for the area in x.x.x.x format. This is the identifier that each neighbor must accept to be part of the same area.
2. On the **Type** tab, select one of the following from the area **Type** list:
 - **Normal**—There are no restrictions; the area can carry all types of routes.
 - **Stub**—There is no outlet from the area. To reach a destination outside of the area, it is necessary to go through the border, which connects to other areas. If you select this option, configure the following:
 - **Accept Summary**—Link state advertisements (LSA) are accepted from other areas. If this option on a stub area Area Border Router (ABR) interface is disabled, the OSPF area will behave as a Totally Stubby Area (TSA) and the ABR will not propagate any summary LSAs.
 - **Advertise Default Route**—Default route LSAs will be included in advertisements to the stub area along with a configured metric value in the configured range 1-255.
 - **NSSA (Not-So-Stubby Area)**—The firewall can leave the area only by routes other than OSPF routes. If you select NSSA, select **Accept Summary** and **Advertise Default Route** as described for **Stub**. If you select this option, configure the following:
 - **Type**—Select either **Ext 1** or **Ext 2** route type to advertise the default LSA.
 - **Ext Ranges**—Add ranges of external routes that you want to **Advertise** or for which you want to **Suppress** advertising.
3. Click **OK**.

STEP 4 | Configure Areas - Range for the OSPF protocol

1. On the **Range** tab, **Add** aggregate LSA destination addresses in the area into subnets.
2. **Advertise** or **Suppress** advertising LSAs that match the subnet, and click **OK**. Repeat to add additional ranges.

STEP 5 | Configure Areas - Interfaces for the OSPF protocol

1. On the **Interface** tab, **Add** the following information for each interface to be included in the area:
 - **Interface**—Select an interface.
 - **Enable**—Selecting this option causes the OSPF interface settings to take effect.
 - **Passive**—Select if you do not want the OSPF interface to send or receive OSPF packets. Although OSPF packets are not sent or received if you choose this option, the interface is included in the LSA database.
 - **Link type**—Choose **Broadcast** if you want all neighbors that are accessible through the interface to be discovered automatically by multicasting OSPF hello messages, such as an Ethernet interface. Choose **p2p** (point-to-point) to automatically discover the neighbor. Choose **p2mp** (point-to-multipoint) when neighbors must be defined manually and **Add** the neighbor IP addresses for all neighbors that are reachable through this interface.
 - **Metric**—Enter an OSPF metric for this interface (range is 0-65,535; default is 10).
 - **Priority**—Enter an OSPF priority for this interface. This is the priority for the router to be elected as a designated router (DR) or as a backup DR (BDR) (range is 0-255; default is 1). If zero is configured, the router will not be elected as a DR or BDR.
 - **Auth Profile**—Select a previously-defined authentication profile.
 - **Timing**—Modify the timing settings if desired (**not recommended**). For details on these settings, refer to the online help.
2. Click **OK**.

STEP 6 | Configure Areas - Virtual Links.

1. On the **Virtual Link** tab, **Add** the following information for each virtual link to be included in the backbone area:
 - **Name**—Enter a name for the virtual link.
 - **Enable**—Select to enable the virtual link.
 - **Neighbor ID**—Enter the router ID of the router (neighbor) on the other side of the virtual link.
 - **Transit Area**—Enter the area ID of the transit area that physically contains the virtual link.
 - **Timing**—It is recommended that you keep the default timing settings.
 - **Auth Profile**—Select a previously-defined authentication profile.
2. Click **OK** to save virtual links.
3. Click **OK** to save area.

STEP 7 | **(Optional)** Configure Auth Profiles.

By default, the firewall does not use OSPF authentication for the exchange between OSPF neighbors. Optionally, you can configure OSPF authentication between OSPF neighbors by

either a simple password or using MD5 authentication. MD5 authentication is recommended; it is more secure than a simple password.

Simple Password OSPF authentication

1. Select the **Auth Profiles** tab and **Add** a name for the authentication profile to authenticate OSPF messages.
2. Select **Simple Password** as the **Password Type**.
3. Enter a simple password and then confirm.

MD5 OSPF authentication

1. Select the **Auth Profiles** tab and **Add** a name for the authentication profile to authenticate OSPF messages.
2. Select **MD5** as the **Password Type** and **Add** one or more password entries, including:
 - Key-ID (range is 0-255)
 - Key
 - Select the **Preferred** option to specify that the key be used to authenticate outgoing messages.
3. Click **OK**.

STEP 8 | Configure Advanced OSPF options.

1. On the **Advanced** tab, select **RFC 1583 Compatibility** to ensure compatibility with RFC 1583.
2. Specify a value for the **SPF Calculation Delay (sec)** timer, which allows you to tune the delay time (in seconds) between receiving new topology information and performing an SPF calculation. Lower values enable faster OSPF re-convergence. Routers peering with the firewall should use the same delay value to optimize convergence times.
3. Specify a value for the **LSA Interval (sec)** timer, which is the minimum time between transmissions of two instances of the same LSA (same router, same type, same LSA ID). This is equivalent to MinLSInterval in RFC 2328. Lower values can be used to reduce re-convergence times when topology changes occur.
4. Click **OK**.

STEP 9 | Commit your changes.

Configure OSPFv3

OSPF supports both IPv4 and IPv6. You must use [OSPFv3](#) if you are using IPv6.

STEP 1 | Configure general [virtual router](#) settings.

STEP 2 | Configure general OSPFv3 configuration settings.

1. Select the **OSPFv3** tab.
2. Select **Enable** to enable the OSPF protocol.
3. Enter the **Router ID**.
4. Select **Reject Default Route** if you do not want to learn any default routes through OSPFv3. This is the recommended default setting.

Clear **Reject Default Route** if you want to permit redistribution of default routes through OSPFv3.

STEP 3 | Configure Auth Profile for the OSPFv3 protocol.

While OSPFv3 doesn't include any authentication capabilities of its own, it relies entirely on IPSec to secure communications between neighbors.

When configuring an authentication profile, you must use Encapsulating Security Payload (ESP) (recommended) or IPv6 Authentication Header (AH).

ESP OSPFv3 authentication

1. On the **Auth Profiles** tab, **Add** a name for the authentication profile to authenticate OSPFv3 messages.
2. Specify a Security Policy Index (**SPI**) (hexadecimal value in the range from 00000000 to FFFFFFFF). The two ends of the OSPFv3 adjacency must have matching SPI values.
3. Select **ESP** for **Protocol**.
4. Select a **Crypto Algorithm**.
You can select **None** or one of the following algorithms: **SHA1**, **SHA256**, **SHA384**, **SHA512**, or **MD5**.
5. If a **Crypto Algorithm** other than None was selected, enter a value for **Key** and then confirm.

AH OSPFv3 authentication

1. On the **Auth Profiles** tab, **Add** a name for the authentication profile to authenticate OSPFv3 messages.
2. Specify a Security Policy Index (**SPI**). The SPI must match between both ends of the OSPFv3 adjacency. The SPI number must be a hexadecimal value between 00000000 and FFFFFFFF.
3. Select **AH** for **Protocol**.
4. Select a **Crypto Algorithm**.
You must enter one of the following algorithms: **SHA1**, **SHA256**, **SHA384**, **SHA512**, or **MD5**.
5. Enter a value for **Key** and then confirm.
6. Click **OK**.
7. Click **OK** again in the Virtual Router - OSPF Auth Profile dialog.

STEP 4 | Configure Areas - Type for the OSPFv3 protocol.

1. On the **Areas** tab, **Add an Area ID**. This is the identifier that each neighbor must accept to be part of the same area.
2. On the **General** tab, select one of the following from the **area Type** list:
 - **Normal**—There are no restrictions; the area can carry all types of routes.
 - **Stub**—There is no outlet from the area. To reach a destination outside of the area, it is necessary to go through the border, which connects to other areas. If you select this option, configure the following:
 - **Accept Summary**—Link state advertisements (LSA) are accepted from other areas. If this option on a stub area Area Border Router (ABR) interface is disabled,

the OSPF area will behave as a Totally Stubby Area (TSA) and the ABR will not propagate any summary LSAs.

- **Advertise Default Route**—Default route LSAs will be included in advertisements to the stub area along with a configured metric value in the configured range 1-255.
- **NSSA (Not-So-Stubby Area)**—The firewall can leave the area only by routes other than OSPF routes. If selected, configure **Accept Summary** and **Advertise Default Route** as described for **Stub**. If you select this option, configure the following:
 - **Type**—Select either **Ext 1** or **Ext 2** route type to advertise the default LSA.
 - **Ext Ranges**—Add ranges of external routes that you want to enable or suppress advertising for.

STEP 5 | Associate an OSPFv3 authentication profile to an area or an interface.

To an Area

1. On the **Areas** tab, select an existing area from the table.
2. On the **General** tab, select a previously defined **Authentication Profile** from the **Authentication** list.
3. Click **OK**.

To an Interface

1. On the **Areas** tab, select an existing area from the table.
2. Select the **Interface** tab and **Add** the authentication profile you want to associate with the OSPF interface from the **Auth Profile** list.
3. Click **OK**.

STEP 6 | Click **OK** again to save the area settings.

STEP 7 | **(Optional)** Configure Export Rules.

1. On the **Export Rules** tab, select **Allow Redistribute Default Route** to permit redistribution of default routes through OSPFv3.
2. Click **Add**.
3. Enter the **Name**; the value must be a valid IPv6 subnet or valid redistribution profile name.
4. Select **New Path Type, Ext 1 or Ext 2**.
5. Specify a **New Tag** for the matched route, using has a 32-bit value in dotted-decimal notation.
6. Assign a **Metric** to the new rule (range is 1-16,777,215).
7. Click **OK**.

STEP 8 | Configure Advanced OSPFv3 options.

1. On the **Advanced** tab, select **Disable Transit Routing for SPF Calculation** if you want the firewall to participate in OSPF topology distribution without being used to forward transit traffic.
2. Specify a value for the **SPF Calculation Delay (sec)** timer, which allows you to tune the delay time (in seconds) between receiving new topology information and performing an

SPF calculation. Lower values enable faster OSPF re-convergence. Routers peering with the firewall should use the same delay value to optimize convergence times.

3. Specify a value for the **LSA Interval (sec)** timer, which is the minimum time (in seconds) between transmissions of two instances of the same LSA (same router, same type, same LSA ID). This is equivalent to MinLSInterval in RFC 2328. Lower values can be used to reduce re-convergence times when topology changes occur.
4. **(Optional) Configure OSPF Graceful Restart.**
5. Click **OK**.

STEP 9 | Commit your changes.

Configure OSPF Graceful Restart

OSPF Graceful Restart directs OSPF neighbors to continue using routes through a firewall during a short transition when it is out of service. This behavior increases network stability by reducing the frequency of routing table reconfiguration and the related route flapping that can occur during short periodic down times.

For a Palo Alto Networks[®] firewall, OSPF Graceful Restart involves the following operations:

- **Firewall as a restarting device**—If the firewall will be down for a short period of time or is unavailable for short intervals, it sends Grace LSAs to its OSPF neighbors. The neighbors must be configured to run in Graceful Restart helper mode. In helper mode, the neighbor receives Grace LSAs informing it that the firewall will perform a graceful restart within a specified period of time defined as the **Grace Period**. During the grace period, the neighbor continues to forward routes through the firewall and to send LSAs that announce routes through the firewall. If the firewall resumes operation before expiration of the grace period, traffic forwarding will continue as before without network disruption. If the firewall does not resume operation after the grace period has expired, the neighbors will exit helper mode and resume normal operation, which will involve reconfiguring the routing table to bypass the firewall.
- **Firewall as a Graceful Restart Helper**—If neighboring routers may be down for short periods of time, the firewall can be configured to operate in Graceful Restart helper mode, in which case the firewall employs a **Max Neighbor Restart Time**. When the firewall receives the Grace LSAs from its OSPF neighbor, it continues to route traffic to the neighbor and advertise routes through the neighbor until either the grace period or max neighbor restart time expires. If neither expires before the neighbor returns to service, traffic forwarding continues as before without network disruption. If either period expires before the neighbor returns to service, the firewall exits helper mode and resumes normal operation, which involves reconfiguring the routing table to bypass the neighbor.

STEP 1 | Select **Network > Virtual Routers** and select the virtual router you want to configure.

STEP 2 | Select **OSPF > Advanced** or **OSPFv3 > Advanced**.

STEP 3 | Verify that the following are selected (they are enabled by default):

- **Enable Graceful Restart**
- **Enable Helper Mode**
- **Enable Strict LSA Checking**

These should remain selected unless required by your topology.

STEP 4 | Configure a **Grace Period** in seconds.

STEP 5 | Configure a **Max Neighbor Restart Time** in seconds.

Confirm OSPF Operation

Once an OSPF configuration has been committed, you can use any of the following operations to confirm that OSPF is operating:

- [View the Routing Table](#)
- [Confirm OSPF Adjacencies](#)
- [Confirm that OSPF Connections are Established](#)

View the Routing Table

By viewing the routing table, you can see whether OSPF routes have been established. The routing table is accessible from either the web interface or the CLI. If you are using the CLI, use the following commands:

- `show routing route`
- `show routing fib`

If you are using the web interface to view the routing table, use the following workflow:

- STEP 1 |** Select **Network > Virtual Routers** and in the same row as the virtual router you are interested in, click the **More Runtime Stats** link.
- STEP 2 |** Select **Routing > Route Table** and examine the **Flags** column of the routing table for routes that were learned by OSPF.

Confirm OSPF Adjacencies

Use the following workflow to confirm that OSPF adjacencies have been established:

- STEP 1 |** Select **Network > Virtual Routers** and in the same row as the virtual router you are interested in, click the **More Runtime Stats** link.
- STEP 2 |** Select **OSPF > Neighbor** and examine the **Status** column to determine if OSPF adjacencies have been established.

Confirm that OSPF Connections are Established

View the System log to confirm that the firewall has established OSPF connections.

- STEP 1 |** Select **Monitor > System** and look for messages to confirm that OSPF adjacencies have been established.
- STEP 2 |** Select **OSPF > Neighbor** and examine the **Status** column to determine if OSPF adjacencies have been established (are full).

BGP

Border Gateway Protocol (BGP) is the primary Internet routing protocol. BGP determines network reachability based on IP prefixes that are available within autonomous systems (AS), where an AS is a set of IP prefixes that a network provider has designated to be part of a single routing policy.

- [BGP Overview](#)
- [MP-BGP](#)
- [Configure BGP](#)
- [Configure a BGP Peer with MP-BGP for IPv4 or IPv6 Unicast](#)
- [Configure a BGP Peer with MP-BGP for IPv4 Multicast](#)
- [BGP Confederations](#)

BGP Overview

BGP functions between autonomous systems (exterior BGP or eBGP) or within an AS (interior BGP or iBGP) to exchange routing and reachability information with BGP speakers. The firewall provides a complete BGP implementation, which includes the following features:

- Specification of one BGP routing instance per virtual router.
- BGP settings per virtual router, which include basic parameters such as local router ID and local AS, and advanced options such as path selection, route reflector, [BGP Confederations](#), route flap dampening, and graceful restart.
- Peer group and neighbor settings, which include neighbor address and remote AS, and advanced options such as neighbor attributes and connections.
- Route policies to control route import, export and advertisement; prefix-based filtering; and address aggregation.
- IGP-BGP interaction to inject routes to BGP using redistribution profiles.
- Authentication profiles, which specify the MD5 authentication key for BGP connections. Authentication helps prevent route leaking and successful DoS attacks.
- Multiprotocol BGP (MP-BGP) to allow BGP peers to carry IPv6 unicast routes and IPv4 multicast routes in Update packets, and to allow the firewall and a BGP peer to communicate with each other using IPv6 addresses.
- BGP supports a maximum of 255 AS numbers in an AS_PATH list for a prefix.

MP-BGP

BGP supports IPv4 unicast prefixes, but a BGP network that uses IPv4 multicast routes or IPv6 unicast prefixes needs multiprotocol BGP (MP-BGP) in order to exchange routes of address types other than IPv4 unicast. MP-BGP allows BGP peers to carry IPv4 multicast routes and IPv6 unicast routes in Update packets, in addition to the IPv4 unicast routes that BGP peers can carry without MP-BGP enabled.

In this way, MP-BGP provides IPv6 connectivity to your BGP networks that use either native IPv6 or dual stack IPv4 and IPv6. Service providers can offer IPv6 service to their customers, and enterprises can use IPv6 service from service providers. The firewall and a BGP peer can communicate with each other using IPv6 addresses.

In order for BGP to support multiple network-layer protocols (other than BGP for IPv4), [Multiprotocol Extensions for BGP-4 \(RFC 4760\)](#) use Network Layer Reachability Information (NLRI) in a Multiprotocol Reachable NLRI attribute that the firewall sends and receives in BGP Update packets. That attribute contains information about the destination prefix, including these two identifiers:

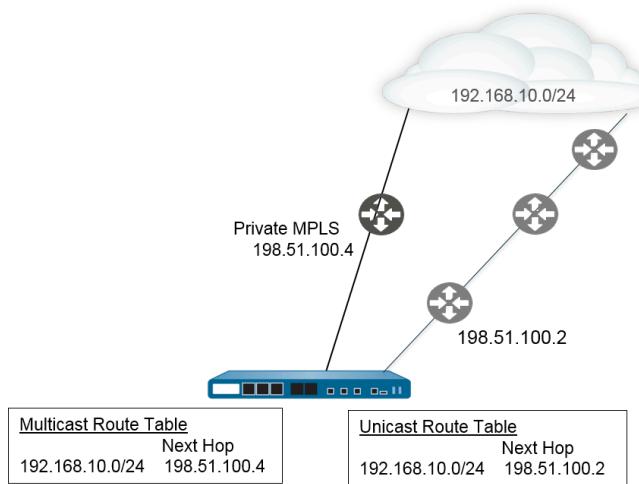
- The Address Family Identifier (AFI), as defined by the IANA in [Address Family Numbers](#), indicates that the destination prefix is an IPv4 or IPv6 address. (PAN-OS supports IPv4 and IPv6 AFIs.)
- The Subsequent Address Family Identifier (SAFI) in PAN-OS indicates that the destination prefix is a unicast or multicast address (if the AFI is IPv4), or that the destination prefix is a unicast address (if the AFI is IPv6). PAN-OS does not support IPv6 multicast.

If you enable MP-BGP for IPv4 multicast or if you configure a multicast static route, the firewall supports separate unicast and multicast route tables for static routes. You might want to separate the unicast and multicast traffic going to the same destination. The multicast traffic can take a different path from unicast traffic because, for example, your multicast traffic is critical, so you need it to be more efficient by having it take fewer hops or undergo less latency.

You can also exercise more control over how BGP functions by configuring BGP to use routes from only the unicast or multicast route table (or both) when BGP imports or exports routes, sends conditional advertisements, or performs route redistribution or route aggregation.

You can decide to use a dedicated multicast RIB (route table) by enabling MP-BGP and selecting the Address Family of IPv4 and Subsequent Address Family of multicast or by installing an IPv4 static route in the multicast route table. After you do either of those methods to use the multicast RIB, the firewall uses the multicast RIB for all multicast routing and reverse path forwarding (RPF). If you prefer to use the unicast RIB for all routing (unicast and multicast), you should not enable the multicast RIB by either method.

In the following figure, a static route to 192.168.10.0/24 is installed in the unicast route table, and its next hop is 198.51.100.2. However, multicast traffic can take a different path to a private MPLS cloud; the same static route is installed in the multicast route table with a different next hop (198.51.100.4) so that its path is different.



Using separate unicast and multicast route tables gives you more flexibility and control when you configure these BGP functions:

- Install an IPv4 static route into the unicast or multicast route table, or both, as described in the preceding example. (You can install an IPv6 static route into the unicast route table only).
- Create an Import rule so that any prefixes that match the criteria are imported into the unicast or multicast route table, or both.
- Create an Export rule so that prefixes that match the criteria are exported (sent to a peer) from the unicast or multicast route table, or both.
- Configure a conditional advertisement with a Non Exist filter so that the firewall searches the unicast or multicast route table (or both) to ensure the route doesn't exist in that table, and so the firewall advertises a different route.
- Configure a conditional advertisement with an Advertise filter so that the firewall advertises routes matching the criteria from the unicast or multicast route table, or both.
- Redistribute a route that appears in the unicast or multicast route table, or both.
- Configure route aggregation with an advertise filter so that aggregated routes to be advertised come from the unicast or multicast route table, or both.
- Conversely, configure route aggregation with a suppress filter so that aggregated routes that should be suppressed (not advertised) come from the unicast or multicast route table, or both.

When you configure a peer with MP-BGP using an Address Family of IPv6, you can use IPv6 addresses in the Address Prefix and Next Hop fields of an Import rule, Export rule, Conditional Advertisement (Advertise Filter and Non Exist Filter), and Aggregate rule (Advertise Filter, Suppress Filter, and Aggregate Route Attribute).

Configure BGP

Perform the following task to configure BGP.

STEP 1 | Configure general [virtual router](#) settings.

STEP 2 | Enable BGP for the virtual router, assign a router ID, and assign the virtual router to an AS.

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **BGP**.
3. **Enable** BGP for this virtual router.
4. Assign a **Router ID** to BGP for the virtual router, which is typically an IPv4 address to ensure the Router ID is unique.
5. Assign the **AS Number**—the number of the AS to which the virtual router belongs based on the router ID (range is 1 to 4,294,967,295).
6. Click **OK**.

STEP 3 | Configure general BGP configuration settings.

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **BGP > General**.
3. Select **Reject Default Route** to ignore any default routes that are advertised by BGP peers.
4. Select **Install Route** to install BGP routes in the global routing table.
5. Select **Aggregate MED** to enable route aggregation even when routes have different Multi-Exit Discriminator (MED) values.
6. Specify the **Default Local Preference** that can be used to determine preferences among different paths.
7. Select the **AS Format** for interoperability purposes:
 - **2 Byte** (default)
 - **4 Byte**



Runtime stats display BGP 4-byte AS numbers using asplain notation according to [RFC 5396](#).

8. Enable or disable each of the following settings for **Path Selection**:
 - **Always Compare MED**—Enable this comparison to choose paths from neighbors in different autonomous systems.
 - **Deterministic MED Comparison**—Enable this comparison to choose between routes that are advertised by IBGP peers (BGP peers in the same autonomous system).
9. For **Auth Profiles**, **Add** an authentication profile:
 - **Profile Name**—Enter a name to identify the profile.
 - **Secret/Confirm Secret**—Enter and confirm a passphrase for BGP peer communications. The Secret is used as a key in MD5 authentication.
10. Click **OK** twice.

STEP 4 | (Optional) Configure BGP settings.

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **BGP > Advanced**.
3. Select **ECMP Multiple AS Support** if you configured ECMP and you want to run ECMP over multiple BGP autonomous systems.
4. **Enforce First AS for EBGP** (enabled by default) to cause the firewall to drop an incoming Update packet from an eBGP peer that does not list the eBGP peer's own AS number as the first AS number in the AS_PATH attribute.
5. Select **Graceful Restart** and configure the following timers:
 - **Stale Route Time (sec)**—Specifies the length of time, in seconds, that a route can stay in the stale state (range is 1 to 3,600; default is 120).
 - **Local Restart Time (sec)**—Specifies the length of time, in seconds, that the local device waits to restart. This value is advertised to peers (range is 1 to 3,600; default is 120).
 - **Max Peer Restart Time (sec)**—Specifies the maximum length of time, in seconds, that the local device accepts as a grace period restart time for peer devices (range is 1 to 3,600; default is 120).
6. For **Reflector Cluster ID**, specify an IPv4 identifier to represent the reflector cluster.
7. For **Confederation Member AS**, specify the autonomous system number identifier (also called a sub-AS number), which is visible only within the BGP confederation. For more information, see [BGP Confederations](#).
8. Add the following information for each Dampening Profile that you want to configure, select **Enable**, and click **OK**:
 - **Profile Name**—Enter a name to identify the profile.
 - **Cutoff**—Specify a route withdrawal threshold above which a route advertisement is suppressed (range is 0.0 to 1,000.0; default is 1.25).
 - **Reuse**—Specify a route withdrawal threshold below which a suppressed route is used again (range is 0.0 to 1,000.0; default is 5).
 - **Max Hold Time (sec)**—Specify the maximum length of time, in seconds, that a route can be suppressed, regardless of how unstable it has been (range is 0 to 3,600; default is 900).
 - **Decay Half Life Reachable (sec)**—Specify the length of time, in seconds, after which a route's stability metric is halved if the route is considered reachable (range is 0 to 3,600; default is 300).
 - **Decay Half Life Unreachable (sec)**—Specify the length of time, in seconds, after which a route's stability metric is halved if the route is considered unreachable (range is 0 to 3,600; default is 300).
9. Click **OK** twice.

STEP 5 | Configure a BGP peer group.

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **BGP > Peer Group**, Add a Name for the peer group, and **Enable** it.
3. Select **Aggregated Confed AS Path** to include a path to the configured aggregated confederation AS.
4. Select **Soft Reset with Stored Info** to perform a soft reset of the firewall after updating the peer settings.
5. Select the **Type** of peer group:
 - **IBGP—Export Next Hop:** Select **Original** or **Use self**.
 - **EBGP Confed—Export Next Hop:** Select **Original** or **Use self**.
 - **EBGP Confed—Export Next Hop:** Select **Original** or **Use self**.
 - **EBGP—Import Next Hop:** Select **Original** or **Use self**; and **Export Next Hop:** Specify **Resolve** or **Use self**. Select **Remove Private AS** if you want to force BGP to remove private AS numbers from the AS_PATH attribute in Updates that the firewall sends to a peer in another AS.
6. Click **OK**.

STEP 6 | Configure a BGP peer that belongs to the peer group and specify its addressing.

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **BGP > Peer Group** and select the peer group you created.
3. For **Peer**, **Add** a peer by **Name**.
4. **Enable** the peer.
5. Enter the **Peer AS** to which the peer belongs.
6. Select **Addressing**.
7. For **Local Address**, select the **Interface** for which you are configuring BGP. If the interface has more than one IP address, enter the IP address for that interface to be the BGP peer.
8. For **Peer Address**, select either **IP** and enter the IP address or select or create an address object, or select **FQDN** and enter the FQDN or address object that is type FQDN.



The firewall uses only one IP address (from each IPv4 or IPv6 family type) from the DNS resolution of the FQDN. If the DNS resolution returns more than one address, the firewall uses the preferred IP address that matches the IP family type (IPv4 or IPv6) configured for the BGP peer. The preferred IP address is the first address the DNS server returns in its initial response. The firewall retains this address as preferred as long as the address appears in subsequent responses regardless of its order.

9. Click **OK**.

STEP 7 | Configure connection settings for the BGP peer.

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **BGP > Peer Group** and select the peer group you created.
3. Select the **Peer** you configured.
4. Select **Connection Options**.
5. Select an **Auth Profile** for the peer.
6. Set a **Keep Alive Interval (sec)**—The interval, in seconds, after which routes from the peer are suppressed according to the Hold Time setting (range is 0 to 1,200; default is 30).
7. Set **Multi Hop**—The time-to-live (TTL) value in the IP header (range is 0 to 255; default is 0). The default value of 0 means 1 for eBGP. The default value of 0 means 255 for iBGP.
8. Set **Open Delay Time (sec)**—The delay, in seconds, between a TCP handshake and the firewall sending the first BGP Open message to establish a BGP connection (range is 0 to 240; default is 0).
9. Set **Hold Time (sec)**—The length of time, in seconds, that may elapse between successive Keepalive or Update messages from the peer before the peer connection is closed (range is 3 to 3,600; default is 90).
10. Set **Idle Hold Time (sec)**—The length of time to wait, in seconds, before retrying to connect to the peer (range is 1 to 3,600; default is 15).
11. Set **Min Route Advertisement Interval (sec)**—The minimum amount of time, in seconds, between two successive Update messages that a BGP speaker (the firewall) sends to a BGP peer that advertise routes or withdrawal of routes (range is 1 to 600; default is 30).
12. For **Incoming Connections**, enter a **Remote Port** and select **Allow** to allow incoming traffic to this port.
13. For **Outgoing Connections**, enter a **Local Port** and select **Allow** to allow outgoing traffic from this port.
14. Click **OK**.

STEP 8 | Configure the BGP peer with settings for route reflector client, peering type, maximum prefixes, and Bidirectional Forwarding Detection (BFD).

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **BGP > Peer Group** and select the peer group you created.
3. Select the **Peer** you configured.
4. Select **Advanced**.
5. For **Reflector Client**, select one of the following:
 - **non-client** (default)—Peer is not a route reflector client.
 - **client**—Peer is a route reflector client.
 - **meshed-client**
6. For **Peering Type**, select one of the following:
 - **Bilateral**—The two BGP peers establish a peer connection.
 - **Unspecified** (default).
7. For **Max Prefixes**, enter the maximum number of IP prefixes to import from the peer (range is 1 to 100,000) or select **unlimited**.
8. To enable **BFD** for the peer (and thereby override the BFD setting for BGP, as long as BFD is not disabled for BGP at the virtual router level), select one of the following:
 - **default**—Peer uses only default BFD settings.
 - **Inherit-vr-global-setting** (default)—Peer inherits the BFD profile that you selected globally for BGP for the virtual router.
 - A BFD profile you configured—See [Create a BFD Profile](#).



Select **Disable BFD** to disable BFD for the BGP peer.

9. Click **OK**.

STEP 9 | Configure Import and Export rules.

The import and export rules are used to import and export routes from and to other routers (for example, importing the default route from your Internet Service Provider).

1. Select **Import**, Add a name (maximum of 63 characters) in the **Rules** field. The name must start with an alphanumeric character and can contain a combination of alphanumeric characters, underscore (_), hyphen (-), dot (.), and space.
2. **Enable** the rule.
3. **Add the Peer Group** from which the routes will be imported.
4. Select **Match** and define the options used to filter routing information. You can also define the Multi-Exit Discriminator (MED) value and a next hop value to routers or

- subnets for route filtering. The MED option is an external metric that lets neighbors know about the preferred path into an AS. A lower value is preferred over a higher value.
5. Select **Action** and define the action that should occur (allow or deny) based on the filtering options defined in the **Match** tab. If you select **Deny**, you don't need to define any additional options. If you select **Allow**, then define the other attributes.
 6. Click **OK**.
 7. Select **Export** and define export attributes, which are similar to the **Import** settings but are used to control route information that is exported from the firewall to neighbors. The name of the Export rule can be a maximum of 31 characters.

STEP 10 | Configure conditional advertising, which allows you to control what route to advertise in the event that a different route is not available in the local BGP routing table (LocRIB), indicating a peering or reachability failure.

This is useful in cases where you want to try to force routes to one AS over another, such as when you have links to the internet through multiple ISPs and you want traffic to be routed to one provider instead of the other except when there is a loss of connectivity to the preferred provider.

1. Select **Conditional Adv** and **Add a Policy** name.
2. **Enable** the conditional advertisement.
3. In the **Used By** section, **Add** the peer groups that will use the conditional advertisement policy.
4. Select **Non Exist Filter** and define the network prefixes of the preferred route. This specifies the route that you want to advertise when it is available in the local BGP routing table. If a prefix is going to be advertised and matches a Non Exist filter, the advertisement will be suppressed.
5. Select **Advertise Filters** and define the prefixes of the route in the Local-RIB routing table that should be advertised in the event that the route in the non-exist filter is unavailable in the local routing table. If a prefix is going to be advertised and does not match a Non Exist filter, the advertisement will occur.
6. Click **OK**.

STEP 11 | Configure aggregate options to summarize routes in the BGP configuration.

BGP route aggregation is used to control how BGP aggregates addresses. Each entry in the table results in the creation of one aggregate address. This will result in an aggregate entry in the routing table when at least one specific route matching the address specified is learned.

1. Select **Aggregate** and **Add** a name for the aggregate address.
2. Enter the network **Prefix** that will be the primary prefix for the aggregated prefixes.
3. Select **Suppress Filters** and define the attributes that will cause the matched routes to be suppressed.
4. Select **Advertise Filters** and define the attributes that will cause the matched routes to always be advertised to peers.
5. Click **OK**.

STEP 12 | Configure redistribution rules.

This rule is used to redistribute host routes and unknown routes that are not on the local RIB to the peer routers.

1. Select **Redist Rules** and **Add** a new redistribution rule.
2. Enter the **Name** of an IP subnet or select a redistribution profile. You can also configure a new redistribution profile if needed.
3. **Enable** the rule.
4. Enter the route **Metric** that will be used for the rule.
5. In the **Set Origin** list, select **incomplete**, **igp**, or **egp**.
6. **(Optional)** Set MED, local preference, AS path limit, and community values.
7. Click **OK**.

STEP 13 | Commit your changes.

Configure a BGP Peer with MP-BGP for IPv4 or IPv6 Unicast

After you [Configure BGP](#), configure a BGP peer with **MP-BGP** for IPv4 or IPv6 unicast for either of the following reasons:

- To have your BGP peer carry IPv6 unicast routes, configure MP-BGP with the Address Family Type of **IPv6** and Subsequent Address Family of **Unicast** so that the peer can send BGP updates that include IPv6 unicast routes. BGP peering (Local Address and Peer Address) can still both be IPv4 addresses, or they can both be IPv6 addresses.
- To perform BGP peering over IPv6 addresses (**Local Address** and **Peer Address** use IPv6 addresses).

The following task shows how to enable a BGP peer with MP-BGP so it can carry IPv6 unicast routes, and so it can peer using IPv6 addresses.

The task also shows how to view the unicast or multicast route tables, and how to view the forwarding table, the BGP local RIB, and BGP RIB Out (routes sent to neighbors) to see routes from the unicast or multicast route table or a specific address family (IPv4 or IPv6).

STEP 1 | Enable MP-BGP Extensions for a peer.

Configure the following so that a BGP peer can carry IPv4 or IPv6 unicast routes in Updates packets and the firewall can use IPv4 or IPv6 addresses to communicate with its peer.

1. Select **Network > Virtual Routers** and select the virtual router you are configuring.
2. Select **BGP**.
3. Select **Peer Group** and select a peer group.
4. Select a BGP peer (router).
5. Select **Addressing**.
6. Select **Enable MP-BGP Extensions** for the peer.
7. For **Address Family Type**, select **IPv4** or **IPv6**. For example, select **IPv6**.
8. For **Subsequent Address Family, Unicast** is selected. If you chose **IPv4** for the Address Family, you can select **Multicast** also.
9. For **Local Address**, select an **Interface** and optionally select an IP address, for example, 2001:DB8:55::/32
10. For **Peer Address**, enter the peer's IP address, using the same address family (IPv4 or IPv6) as the Local Address, for example, 2001:DB8:58::/32.
11. Select **Advanced**.
12. **(Optional) Enable Sender Side Loop Detection.** When you enable Sender Side Loop Detection, the firewall will check the AS_PATH attribute of a route in the BGP RIB before it sends the route in an update, to ensure that the peer AS number isn't in the AS_PATH list. The firewall doesn't advertise the route if the peer AS number is in the AS_PATH list. Usually the receiver detects loops, but this optimization feature has the sender perform the loop detection. Disable this feature to have the receiver perform loop detection.
13. Click **OK**.

STEP 2 | (Optional) Create a static route and install it in the unicast route table because you want the route to be used only for unicast purposes.

1. Select **Network > Virtual Routers** and select the virtual router you are configuring.
2. Select **Static Routes**, select **IPv4** or **IPv6**, and **Add** a route.
3. Enter a **Name** for the static route.
4. Enter the IPv4 or IPv6 **Destination** prefix and netmask, depending on whether you chose IPv4 or IPv6.
5. Select the egress **Interface**.
6. Select the **Next Hop as IPv6 Address** (or **IP Address** if you chose IPv4) and enter the address of the next hop to which you want to direct unicast traffic for this static route.
7. Enter an **Admin Distance**.
8. Enter a **Metric**.
9. For **Route Table**, select **Unicast**.
10. Click **OK**.

STEP 3 | Commit the configuration.

Click **Commit**.

STEP 4 | View the unicast or multicast route table.

1. Select **Network > Virtual Routers**.
2. In the row for the virtual router, click **More Runtime Stats**.
3. Select **Routing > Route Table**.
4. For **Route Table**, select **Unicast** or **Multicast** to display only those routes.
5. For **Display Address Family**, select **IPv4 Only**, **IPv6 Only**, or **IPv4 and IPv6** to display only routes for that address family.



Selecting Multicast with IPv6 Only is not supported.

STEP 5 | View the Forwarding Table.

1. Select **Network > Virtual Routers**.
2. In the row for the virtual router, click **More Runtime Stats**.
3. Select **Routing > Forwarding Table**.
4. For **Display Address Family**, select **IPv4 Only**, **IPv6 Only**, or **IPv4 and IPv6** to display only routes for that address family.

STEP 6 | View the BGP RIB tables.

1. View the BGP Local RIB, which shows the BGP routes that the firewall uses to route BGP packets.
 1. Select **Network > Virtual Routers**.
 2. In the row for the virtual router, click **More Runtime Stats**.
 3. Select **BGP > Local RIB**.
 4. For **Route Table**, select **Unicast** or **Multicast** to display only those routes.
 5. For **Display Address Family**, select **IPv4 Only**, **IPv6 Only**, or **IPv4 and IPv6** to display only routes for that address family.



Selecting Multicast with IPv6 Only is not supported.

2. View the BGP RIB Out table, which shows the routes that the firewall sends to BGP neighbors.
 1. Select **Network > Virtual Routers**.
 2. In the row for the virtual router, click **More Runtime Stats**.
 3. Select **BGP > RIB Out**.
 4. For **Route Table**, select **Unicast** or **Multicast** to display only those routes.
 5. For **Display Address Family**, select **IPv4 Only**, **IPv6 Only**, or **IPv4 and IPv6** to display only routes for that address family.



Selecting Multicast with IPv6 Only is not supported.

Configure a BGP Peer with MP-BGP for IPv4 Multicast

After you [Configure BGP](#), configure a BGP peer with MP-BGP for IPv4 multicast if you want your BGP peer to be able to learn and pass IPv4 multicast routes in BGP updates. You'll be able to separate unicast from multicast traffic, or employ the features listed in [MP-BGP](#) to use only routes from the unicast or multicast route table, or routes from both tables.

If you want to support multicast traffic only, you must use a filter to eliminate unicast traffic.

The firewall doesn't support ECMP for multicast traffic.

STEP 1 | Enable MP-BGP extensions so that a BGP peer can exchange IPv4 multicast routes.

1. Select **Network > Virtual Routers** and select the virtual router you are configuring.
2. Select **BGP**.
3. Select **Peer Group**, select a peer group and a BGP peer.
4. Select **Addressing**.
5. Select **Enable MP-BGP Extensions**.
6. For **Address Family Type**, select **IPv4**.
7. For **Subsequent Address Family**, select **Unicast** and then **Multicast**.
8. Click **OK**.

STEP 2 | (Optional) Create an IPv4 static route and install it in the multicast route table only.

You would do this to direct multicast traffic for a BGP peer to a specific next hop, as shown in the topology in [MP-BGP](#).

1. Select **Network > Virtual Routers** and select the virtual router you are configuring.
2. Select **Static Routes > IPv4** and **Add a Name** for the route.
3. Enter the **IPv4 Destination** prefix and netmask.
4. Select the **Egress Interface**.
5. Select the **Next Hop as IP Address** and enter the IP address of the next hop to which you want to direct multicast traffic for this static route.
6. Enter an **Admin Distance**.
7. Enter a **Metric**.
8. For **Route Table**, select **Multicast**.
9. Click **OK**.

STEP 3 | Commit the configuration.

Click **Commit**.

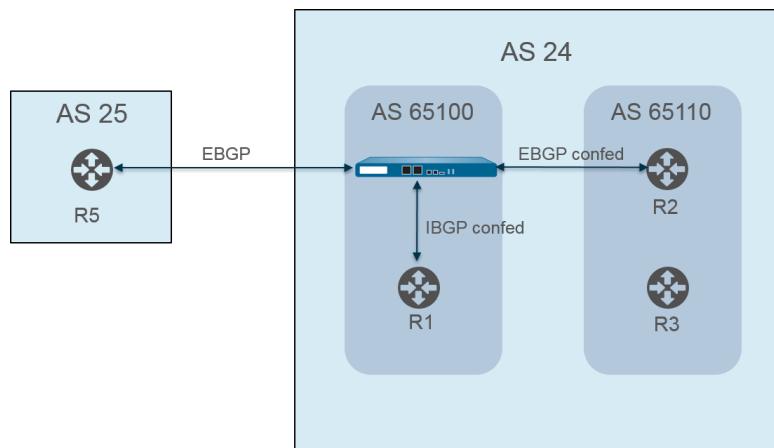
STEP 4 | View the route table.

1. Select **Network > Virtual Routers**.
2. In the row for the virtual router, click **More Runtime Stats**.
3. Select **Routing > Route Table**.
4. For **Route Table**, select **Unicast** or **Multicast** to display only those routes.
5. For **Display Address Family**, select **IPv4 Only**, **IPv6 Only**, or **IPv4 and IPv6** to display only routes for that address family.

STEP 5 | To view the Forwarding table, BGP Local RIB, or BGP RIB Out table, see [Configure a BGP Peer with MP-BGP for IPv4 or IPv6 Unicast](#).

BGP Confederations

BGP confederations provide a way to divide an autonomous system (AS) into two or more sub-autonomous systems (sub-AS) to reduce the burden that the full mesh requirement for IBGP causes. The firewalls (or other routing devices) within a sub-AS must still have a full iBGP mesh with the other firewalls in the same sub-AS. You need BGP peering between sub-autonomous systems for full connectivity within the main AS. The firewalls peering with each other within a sub-AS form an IBGP confederation peering. The firewall in one sub-AS peering with a firewall in a different sub-AS form an EBGP confederation peering. Two firewalls from different autonomous systems that connect are EBGP peers.

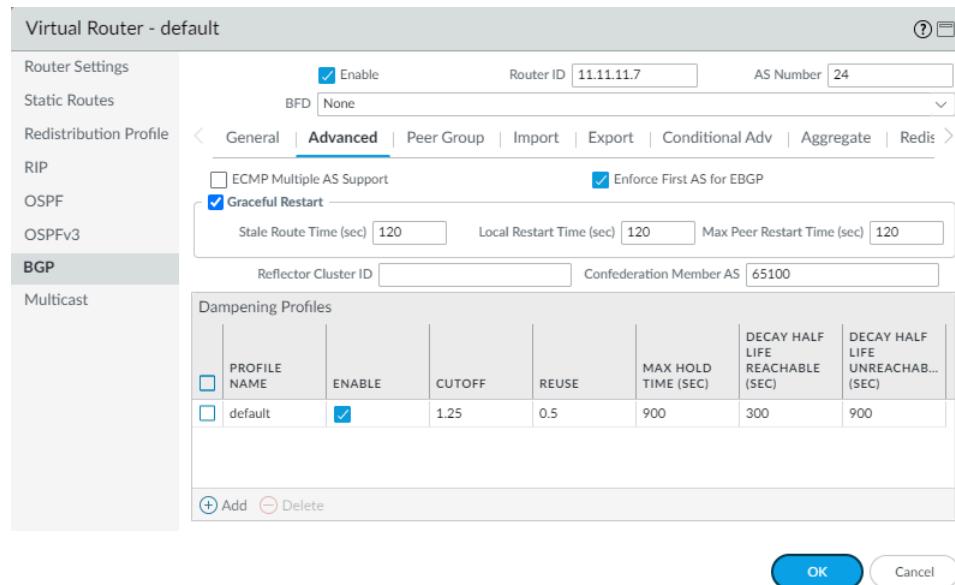


Autonomous systems are identified with a public (globally-assigned) AS number, such as AS 24 and AS 25 in the preceding figure. In a PAN-OS environment, you assign each sub-AS a unique Confederation Member AS number, which is a private number seen only within the AS. In this figure, the confederations are AS 65100 and AS 65110. ([RFC6996](#), Autonomous System (AS) Reservation for Private Use, indicates that the IANA reserves AS numbers 64512-65534 for private use.)

The sub-AS confederations seem like full autonomous systems to each other within the AS. However, when the firewall sends an AS path to an EBGP peer, only the public AS number appears in the AS path; no private sub-AS (Confederation Member AS) numbers are included.

BGP peering occurs between the firewall and R2; the firewall in the figure has these relevant configuration settings:

- AS number—24
- Confederation Member AS—65100
- Peering Type—EBGP confed
- Peer AS—65110



Router 2 (R2) in AS 65110 is configured as follows:

- AS number—24
- Confederation Member AS—65110
- Peering Type—EBGP confed
- Peer AS—65100

BGP peering also occurs between the firewall and R1. The firewall has the following additional configuration:

- AS number—24
- Confederation Member AS—65100
- Peering Type—IBGP confed
- Peer AS—65110

R1 is configured as follows:

- AS number—24
- Confederation Member AS—65110
- Peering Type—IBGP confed
- Peer AS—65100

BGP peering occurs between the firewall and R5. The firewall has the following additional configuration:

- AS number—24
- Confederation Member AS—65100
- Peering Type—EBGP
- Peer AS—25

R5 is configured as follows:

- AS-25
- Peering Type—EBGP
- Peer AS-24

After the firewall is configured to peer with R1, R2, and R5, its peers are visible on the **Peer Group** tab:

			Peers			
	NAME	ENABLE	TYPE	NAME	PEER ADDRESS	LOCAL ADDRESS
<input type="checkbox"/>	iBGP_confed	<input checked="" type="checkbox"/>	ibgp-confed	R1	11.11.11.6	11.11.11.7/24

The firewall shows the R1, R2, and R5 peers:

PEER	ENABLE	PEER AS	LOCAL ADDRESS	PEER ADDRESS	MAX PREFIXES
R1	<input checked="" type="checkbox"/>	65100	11.11.11.7/24	11.11.11.6	5000

Virtual Router - BGP - Peer Group/Peer

Peer Group						
Name		EBGP_confed				
<input checked="" type="checkbox"/> Enable		Type	EBGP Confed			
<input checked="" type="checkbox"/> Aggregated Confed AS Path		Export Next Hop <input checked="" type="radio"/> Original <input type="radio"/> Use Self				
<input type="checkbox"/> Soft Reset With Stored Info						
<input type="checkbox"/>	PEER	ENABLE	PEER AS	LOCAL ADDRESS	PEER ADDRESS	MAX PREFIXES
<input type="checkbox"/>	R2	<input checked="" type="checkbox"/>	65110	11.11.11.6/24	11.11.11.7	5000

[+ Add](#) [Delete](#)

[OK](#) [Cancel](#)

Virtual Router - BGP - Peer Group/Peer

Peer Group						
Name		EBGP				
<input checked="" type="checkbox"/> Enable		Type	EBGP			
<input checked="" type="checkbox"/> Aggregated Confed AS Path		Import Next Hop <input checked="" type="radio"/> Original <input type="radio"/> Use Peer Export Next Hop <input checked="" type="radio"/> Resolve <input type="radio"/> Use Self <input type="checkbox"/> Remove Private AS				
<input type="checkbox"/> Soft Reset With Stored Info						
<input type="checkbox"/>	PEER	ENABLE	PEER AS	LOCAL ADDRESS	PEER ADDRESS	MAX PREFIXES
<input type="checkbox"/>	R5	<input checked="" type="checkbox"/>	25	11.1.1.1/24	11.1.1.11	5000

[+ Add](#) [Delete](#)

[OK](#) [Cancel](#)

To verify that the routes from the firewall to the peers are established, on the virtual router's screen, select **More Runtime Stats** and select the **Peer** tab.

Virtual Router - virtual_router

Routing | RIP | OSPF | OSPFv3 | **BGP** | Multicast | BFD Summary Information

Summary | **Peer** | Peer Group | Local RIB | RIB Out

NAME	GROUP	LOCAL IP	PEER IP	PEER AS	PASSWORD SET	STATUS	STATUS DURATION (SECS.)
R1	iBGP_confed	12.1.1.1:35636	12.1.1.2:179	65100	no	Established	4281
R2	EBGP_confed	15.1.1.1:179	15.1.1.5:39783	65110	no	Established	1424
R5	EBGP	111.1.1.1:37699	111.1.1.11:179	24	no	Established	769

Close

Select the **Local RIB** tab to view information about the routes stored in the Routing Information Base (RIB).

Virtual Router - virtual_router

Routing | RIP | OSPF | OSPFv3 | **BGP** | Multicast | BFD Summary Information

Summary | Peer | Peer Group | **Local RIB** | RIB Out

Route Table Unicast Multicast Display Address Family IPv4 and IPv6

PREFIX	FLAG	NEXT HOP	PEER	WEIGHT	LOCAL PREF.	AS PATH	ORIGIN	MED	FLAP COUNT
13.1.1.0/24		222.1.1.11	R1	0	100		N/A	0	0
25.1.1.0/24	*	15.1.1.5	R2	0	100	[65110]	N/A	0	0
3.3.3.0/24	*	46.46.46.4	R5	0	100	25	N/A	0	0

Close

Then select the **RIB Out** tab.

BGP

Virtual Router - virtual_router

Routing | RIP | OSPF | OSPFv3 | **BGP** | Multicast | BFD Summary Information

Summary | Peer | Peer Group | Local RIB | **RIB Out**

Route Table Unicast Multicast Display Address Family IPv4 and IPv6 ▾

PREFIX	NEXT HOP	PEER	LOCAL PREF.	AS PATH	ORIGIN	MED	ADV. STATUS	AGGR. STATUS
3.3.3.0/24	46.46.46.4	R1	100	25	N/A	0	advertised	no aggregate
25.1.1.0/24	15.1.1.5	R1	100	[65110]	N/A	0	advertised	no aggregate
3.3.3.0/24	46.46.46.4	R2	100	[65100],25	N/A	0	advertised	no aggregate
25.1.1.0/24	46.46.46.6	R5	0	26	N/A	0	advertised	no aggregate

4 items → X

Close

IP Multicast

IP multicast is a set of protocols that network appliances use to send multicast IP datagrams to a group of interested receivers using one transmission rather than unicasting the traffic to multiple receivers, thereby saving bandwidth. IP multicast is suitable for communication from one source (or many sources) to many receivers, such as audio or video streaming, IPTV, video conferencing, and distribution of other communication, such as news and financial data.

A multicast address identifies a group of receivers that want to receive the traffic going to that address. You should not use the multicast addresses reserved for special uses, such as the range 224.0.0.0 through 224.0.0.255 or 239.0.0.0 through 239.255.255.255. Multicast traffic uses UDP, which does not resend missed packets.

Palo Alto Networks® firewalls support IP multicast and Protocol Independent Multicast (PIM) on a Layer 3 interface that you configure for a [virtual router](#) on the firewall.

For multicast routing, the Layer 3 interface type can be Ethernet, Aggregate Ethernet (AE), VLAN, loopback, or tunnel. Interface groups allow you to configure more than one firewall interface at a time with the same Internet Group Management Protocol (IGMP) and PIM parameters, and with the same group permissions (multicast groups allowed to accept traffic from any source or from only a specific source). An interface can belong to only one interface group.

The firewall supports IPv4 multicast—it does not support IPv6 multicast. The firewall also does not support PIM Dense Mode (PIM-DM), IGMP proxy, IGMP static joins, Anycast RP, GRE, or multicast configurations on a Layer 2 or virtual wire interface type. However, a virtual wire interface can pass multicast packets. Also, a Layer 2 interface can switch Layer 3 IPv4 multicast packets between different VLANs and the firewall will retag the VLAN ID using the VLAN ID of the egress interface.

You must enable multicast for a virtual router and enable PIM for an ingress and an egress interface in order for the interfaces to receive or forward multicast packets. In addition to PIM, you must also enable IGMP on egress interfaces that face receivers. You must configure a Security policy rule to allow IP multicast traffic to a predefined Layer 3 destination zone named **multicast** or to **any** destination zone.

- [IGMP](#)
- [PIM](#)
- [Configure IP Multicast](#)
- [View IP Multicast Information](#)

IGMP

Internet Group Management Protocol (IGMP) is an IPv4 protocol that a multicast receiver uses to communicate with an interface on a Palo Alto Networks® firewall and that the firewall uses to track the membership of multicast groups. When a host wants to receive multicast traffic, its implementation of IGMP sends an IGMP Membership report message and the receiving router, in turn, sends a PIM Join message to the multicast group address of the group that the host wants to join. An IGMP-enabled router on the same physical network (such as an Ethernet segment) then uses PIM to communicate with other PIM-enabled routers to determine a path from the source to interested receivers.

Enable IGMP only on interfaces that face a multicast receiver. The receivers can be only one Layer 3 hop away from the virtual router. IGMP messages are Layer 2 messages that have a TTL value of one and, therefore, cannot go outside the LAN.

When you [Configure IP Multicast](#), specify whether an interface uses [IGMP Version 1](#), [IGMP Version 2](#), or [IGMP Version 3](#). You can enforce the IP Router Alert option, [RFC 2113](#), so that incoming IGMP packets that use IGMPv2 or IGMPv3 have the IP Router Alert option.

By default, an interface accepts IGMP Membership reports for all multicast groups. You can configure multicast group permissions to control the groups for which the virtual router accepts Membership reports from any source (Any-Source Multicast, or ASM), which is basically PIM Sparse Mode (PIM-SM). You can also specify the groups for which the virtual router accepts Membership reports from a specific source (PIM Source-Specific Multicast [PIM-SSM]). If you specify permissions for either ASM or SSM groups, the virtual router denies Membership reports from other groups. The interface must use IGMPv3 to pass PIM-SSM traffic.

You can specify the maximum number of sources and the maximum number of multicast groups that IGMP can process simultaneously for an interface.

The virtual router multicasts an IGMP Query at regular intervals to all receivers of a multicast group. A receiver responds to an IGMP Query with an IGMP Membership report that confirms the receiver still wants to receive multicast traffic for that group. The virtual router maintains a table of the multicast groups that have receivers; the virtual router forwards a multicast packet out the interface to the next hop only if there is still a receiver down that multicast distribution tree that is joined to the group. The virtual router does not track exactly which receivers are joined to a group. Only one router on a subnet responds to IGMP Queries and that is the IGMP Querier—the router with the lowest IP address.

You can configure an interface with an IGMP Query interval and the amount of time allowed for a receiver to respond to a query (the Max Query Response Time). When a virtual router receives an IGMP Leave message from a receiver to leave a group, the virtual router checks that the interface that received the Leave message is not configured with the Immediate Leave option. In the absence of the Immediate Leave option, the virtual router sends a Query to determine whether there are still receiver members for the group. The Last Member Query Interval specifies how many seconds are allowed for any remaining receivers for that group to respond and confirm that they still want multicast traffic for that group.

An interface supports the IGMP robustness variable, which you can adjust so that the firewall then tunes the Group Membership Interval, Other Querier Present Interval, Startup Query Count, and Last Member Query Count. A higher robustness variable can accommodate a subnet that is likely to drop packets.

[View IP Multicast Information](#) to see IGMP-enabled interfaces, the IGMP version, Querier address, robustness setting, limits on the number of multicast groups and sources, and whether the interface is configured for Immediate Leave. You can also see the multicast groups to which interfaces belong and other IGMP membership information.

PIM

IP multicast uses the Protocol Independent Multicast (PIM) routing protocol between routers to determine the path on the distribution tree that multicast packets take from the source to the receivers (multicast group members). Both virtual routers (on a legacy routing engine) and logical routers (on an Advanced Routing Engine) support PIM.

A Palo Alto Networks® firewall supports PIM Sparse Mode (PIM-SM) ([RFC 4601](#)), PIM Any-Source Multicast (ASM) (sometimes referred to as PIM Sparse Mode), and PIM Source-Specific Multicast (SSM). In PIM-SM, the source does not forward multicast traffic until a receiver (user) belonging to a multicast group requests that the source send the traffic. When a host wants to receive multicast traffic, its implementation of IGMP sends an IGMP Membership report message, and the receiving router then sends a PIM Join message to the multicast group address of the group it wants to join.

- In **ASM**, the receiver uses IGMP to request traffic for a multicast group address; any source could have originated that traffic. Consequently, the receiver doesn't necessarily know the senders, and the receiver could receive multicast traffic in which it has no interest.
- In **SSM** ([RFC 4607](#)), the receiver uses IGMP to request traffic from one or more specific sources to a multicast group address. The receiver knows the IP address of the senders and receives only the multicast traffic it wants. SSM requires IGMPv3. The default SSM address space (232.0.0.0/8) can be overridden by adjusting the [source specific address space](#). [Group permissions](#) also need to be adjusted.

When you [Configure IP Multicast](#) on a Palo Alto Networks firewall, you must enable PIM for an interface to forward multicast traffic, even on receiver-facing interfaces. This is unlike IGMP, which you enable only on receiver-facing interfaces.

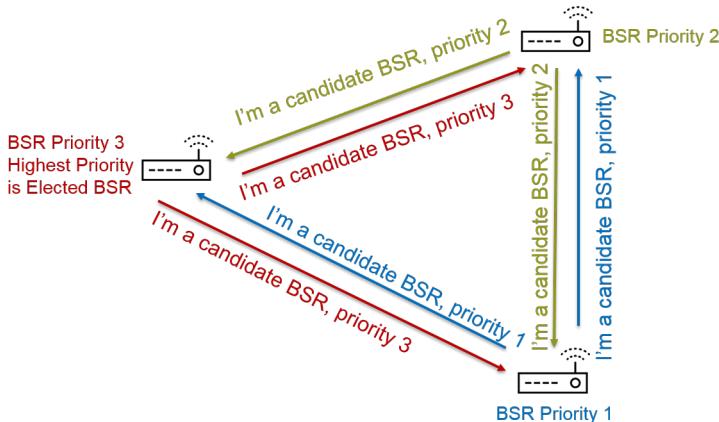
ASM requires a *rendezvous point* (RP), which is a router located at the juncture or root of a shared distribution tree. The RP for a multicast domain serves as a single point to which all multicast group members send their Join messages. This behavior reduces the likelihood of a routing loop that would otherwise occur if group members sent their Join messages to multiple routers. (SSM doesn't need an RP because source-specific multicast uses a shortest-path tree and therefore has no need for an RP.)

In an ASM environment, there are two ways that the virtual router determines which router is the RP for a multicast group:

- **Static RP-to-Group Mapping**—configures the virtual router on the firewall to act as RP for multicast groups. You configure a local RP, either by configuring a static RP address or by specifying that the local RP is a candidate RP and the RP is chosen dynamically (based on lowest priority value). You can also statically configure one or more external RPs for different group address ranges not covered by the local RP, which helps you load-balance multicast traffic so that one RP is not overloaded.

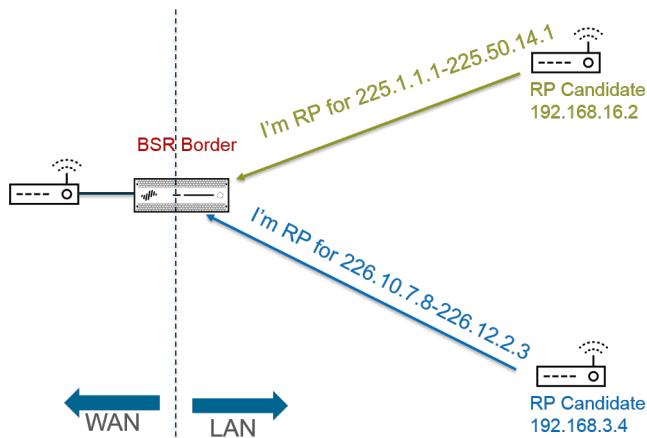
- **Bootstrap Router (BSR)**—(RFC 5059)—defines the role of a BSR. First, candidates for BSR advertise their priority to each other and then the candidate with the largest priority is elected BSR, as shown in the following figure:

RPs Advertise Their BSR Candidacy; Highest Priority Wins



Next, the BSR discovers RPs when candidate RPs periodically unicast a BSR message to the BSR containing their IP address and the multicast group range for which they will act as RP. You can configure the local virtual router to be a candidate RP, in which case the virtual router announces its RP candidacy for a specific multicast group or groups. The BSR sends out RP information to the other RPs in the PIM domain.

When you configure PIM for an interface, you can select BSR Border when the interface on the firewall is at an enterprise boundary facing away from the enterprise network. The BSR Border setting prevents the firewall from sending RP candidacy BSR messages outside the LAN. In the following illustration, BSR Border is enabled for the interface facing the LAN and that interface has the highest priority. If the virtual router has both a static RP and a dynamic RP (learned from the BSR), you can specify whether the static RP should override the learned RP for a group when you configure the local, static RP.

BSR Border Router Discovers RPs;
Keeps PIM RP Candidacy Messages Within LAN

In order for PIM Sparse Mode to notify the RP that it has traffic to send down a shared tree, the RP must be aware of the source. The host notifies the RP that it is sending traffic to a multicast

group address when the *designated router* (DR) encapsulates the first packet from the host in a PIM Register message and unicasts the packet to the RP on its local network. The DR also forwards Prune messages from a receiver to the RP. The RP maintains the list of IP addresses of sources that are sending to a multicast group and the RP can forward multicast packets from sources.

Why do the routers in a PIM domain need a DR? When a router sends a PIM Join message to a switch, two routers could receive it and forward it to the same RP, causing redundant traffic and wasting bandwidth. To prevent unnecessary traffic, the PIM routers elect a DR (the router with the highest IP address), and only the DR forwards the Join message to the RP. Alternatively, you can assign a DR priority to an interface group, which takes precedence over IP address comparisons. As a reminder, the DR is forwarding (unicasting) PIM messages; it is not multicasting IP multicast packets.

You can specify the IP addresses of PIM neighbors (routers) that the interface group will allow to peer with the virtual router. By default, all PIM-enabled routers can be PIM neighbors, but the option to limit neighbors provides a step toward securing the virtual router in your PIM environment.

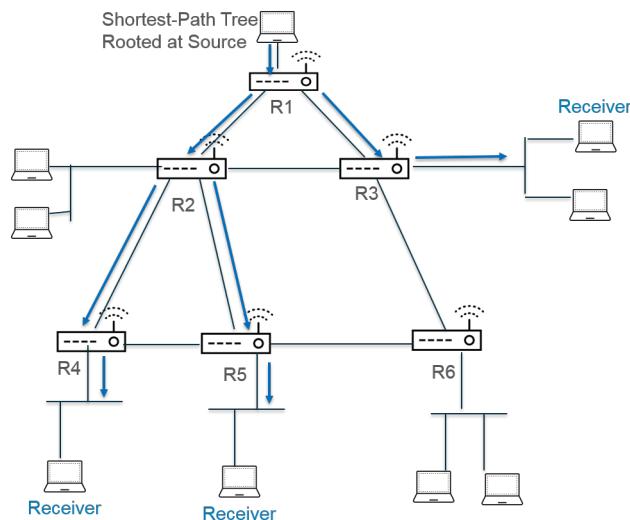
- [Shortest Path Tree \(SPT\) and Shared Tree](#)
- [PIM Assert Mechanism](#)
- [Reverse-Path Forwarding](#)

Shortest-Path Tree (SPT) and Shared Tree

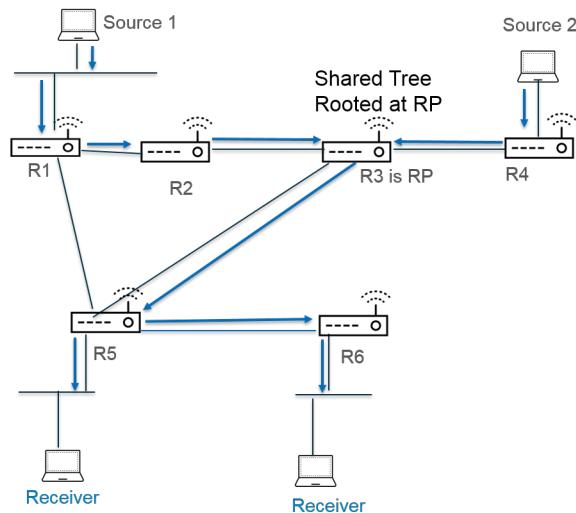
After a receiver joins a multicast group, the routers in the multiaccess network build the routing paths necessary to send data to each receiver in the group. Each IP datagram sent to a multicast group is distributed (forwarded) to all members. The routing paths constitute a type of distribution tree for a multicast packet. The goal of a multicast distribution tree is for the router to duplicate a multicast packet when the packet reaches a divergence of paths and the router must send the packet down multiple paths to reach all group members, yet the distribution tree must refrain from sending packets down a path where no interested receivers exist. The distribution tree is one of the following:

- A **source tree**—A path from a multicast source (the root of the tree) through the network to the receivers in the multicast group. The source tree is the shortest path that a multicast packet can take from source to receiver, so it is also known as the **shortest-path tree (SPT)**. The sender and receiver are annotated as a source and multicast group pair, shortened to (S,

G); for example, (192.168.1.1, 225.9.2.6). The following figure illustrates three shortest-path trees from the source to three receivers.



- A **shared tree**—A path rooted at the RP, not at the multicast source. A shared tree is also known as an RP tree or RPT. Routers forward multicast packets from various sources to the RP and the RP forwards the packets down the shared tree. A shared tree is annotated as (*, G), using a wildcard as the source because all sources belonging to the multicast group share the same distribution tree from the RP. An example shared tree annotation is (*, 226.3.1.5). The following figure illustrates a shared tree from the root at the RP to the receivers.



[Source-Specific Multicast \(SSM\)](#) uses source tree distribution. When you [Configure IP Multicast](#) to use Any Source Multicast (ASM), you can specify which distribution tree the virtual router on your Palo Alto Networks® firewall uses to deliver multicast packets to a group by setting an SPT threshold for the group:

- By default the virtual router switches multicast routing from shared tree to SPT when it receives the first multicast packet for a group or prefix (the **SPT Threshold** is set to 0).

- You can configure the virtual router to switch to SPT when the total number of kilobits in packets arriving for the specified multicast group or prefix at any interface over any length of time reaches a configured number.
- You can configure the virtual router to never switch to SPT for the group or prefix (it continues to use shared tree).

SPT requires more memory, so choose your setting based on your multicast traffic level to the group. If the virtual router switches to SPT, then packets will arrive from the source (rather than the RP) and the virtual router sends a Prune message to the RP. The source sends subsequent multicast packets for that group down the shortest-path tree.

PIM Assert Mechanism

To prevent routers on a multiaccess network from forwarding the same multicast traffic to the same next hop (which would cause redundant traffic and wasted bandwidth), PIM uses the Assert mechanism to elect a single PIM Forwarder for the multiaccess network.

If the virtual router receives a multicast packet from a source on an interface that the virtual router already associates as the outgoing interface for the same (S,G) pair identified in the packet, that means this is a duplicate packet. Consequently, the virtual router sends an Assert message containing its metrics to the other routers on the multiaccess network. The routers then elect a PIM Forwarder in this manner:

1. The PIM Forwarder is the router with the lowest administrative distance to the multicast source.
2. In the event of a tie for lowest administrative distance, the PIM Forwarder is the router with the best unicast routing metric to the source.
3. In the event of a tie for best metric, the PIM Forwarder is the router with the highest IP address.

Routers that are not elected as the PIM Forwarder will stop forwarding traffic to the multicast group identified in the (S,G) pair.

When you [Configure IP Multicast](#), you can configure the interval at which the virtual router sends PIM Assert messages out an interface (the Assert interval). When you [View IP Multicast Information](#), the **PIM Interface** tab displays the Assert interval for an interface.

Reverse-Path Forwarding

PIM uses reverse-path forwarding (RPF) to prevent multicast routing loops by leveraging the unicast routing table on the virtual router. When the virtual router receives a multicast packet, it looks up the source of the multicast packet in its unicast routing table to see if the outgoing interface associated with that source IP address is the interface on which that packet arrived. If the interfaces match, the virtual router duplicates the packet and forwards it out the interfaces toward the multicast receivers in the group. If the interfaces don't match, the virtual router drops the packet. The unicast routing table is based on the underlying static routes or the interior gateway protocol (IGP) your network uses, such as OSPF.

PIM also uses RPF to build a [shortest-path tree](#) to a source, one PIM router hop at a time. The virtual router has the address of the multicast source, so the virtual router selects as its next hop back to the source the upstream PIM neighbor that the virtual router would use to forward unicast packets to the source. The next hop router does the same thing.

After RPF succeeds and the virtual router has a route entry in its multicast routing information base (mRIB), the virtual router maintains source-based tree entries (S,G) and shared tree entries (*,G) in its multicast forwarding information base (multicast forwarding table or mFIB). Each entry includes the source IP address, multicast group, incoming interface (RPF interface) and outgoing interface list. There can be multiple outgoing interfaces for an entry because the shortest path tree can branch at the router, and the router must forward the packet out multiple interfaces to reach receivers of the group that are located down different paths. When the virtual router uses the mFIB to forward a multicast packet, it matches an (S,G) entry before it attempts to match a (*,G) entry.

If you are advertising multicast source prefixes into BGP (you configured [MP-BGP](#) with the IPv4 Address Family and the multicast Subsequent Address Family), then the firewall always performs the RPF check on the BGP routes that the firewall received under the multicast Subsequent Address Family.

[View IP Multicast Information](#) to see how to view the mFIB and mRIB entries. Keep in mind that the multicast route table (mRIB) is a separate table from the unicast route table (RIB).

Configure IP Multicast

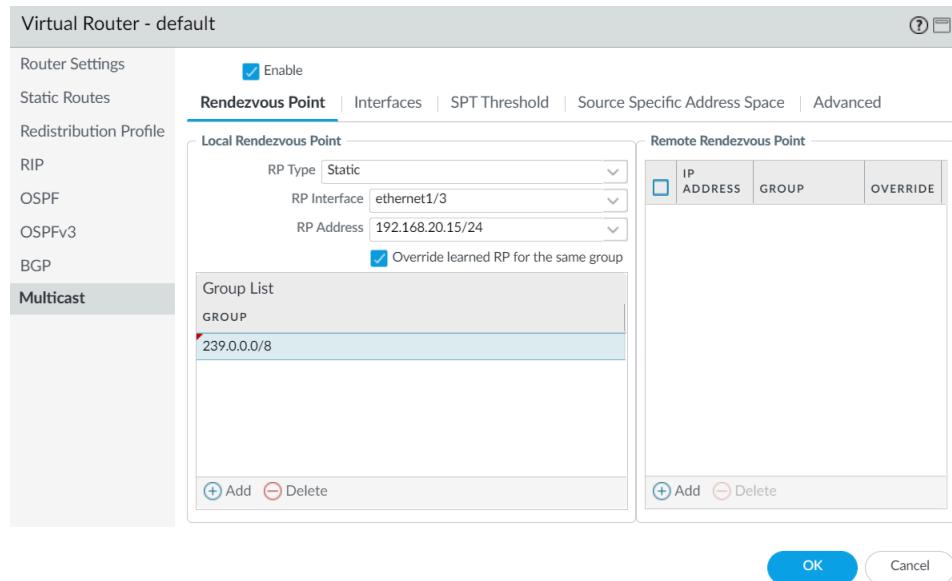
Configure interfaces on a virtual router of a Palo Alto Networks® firewall to receive and forward [IP Multicast](#) packets. You must enable IP multicast for the virtual router, configure Protocol Independent Multicast (PIM) on the ingress and egress interfaces, and configure Internet Group Management Protocol (IGMP) on receiver-facing interfaces.

STEP 1 | Enable IP multicast for a virtual router.

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **Multicast** and **Enable IP multicast**.

STEP 2 | (ASM only) If the multicast domain in which the virtual router is located uses Any-Source Multicast (ASM), identify and configure the local and remote rendezvous points (RPs) for multicast groups.

1. Select **Rendezvous Point**.
2. Select a **Local RP Type**, which determines how the RP is chosen (the options are **Static**, **Candidate**, or **None**):
 - **Static**—Establishes a static mapping of an RP to multicast groups. Configuring a static RP requires you to explicitly configure the same RP on other PIM routers in the PIM domain.
 - Select the **RP Interface**. Valid interface types are Layer3, virtual wire, loopback, VLAN, Aggregate Ethernet (AE), and tunnel.
 - Select the **RP Address**. The IP addresses of the RP interface you selected populate the list.
 - Select **Override learned RP for the same group** so that this static RP serves as RP instead of the RP elected for the groups in the Group List.
 - Add one or more multicast **Groups** for which the RP acts as the RP.



- **Candidate**—Establishes a dynamic mapping of an RP to multicast groups based on priority so that each router in a PIM domain automatically elects the same RP.
 - Select the **RP Interface** of the candidate RP. Valid interface types are Layer 3, loopback, VLAN, Aggregate Ethernet (AE), and tunnel.
 - Select the **RP Address** of the candidate RP. The IP addresses for the RP interface you selected populate the list.
 - (**Optional**) Change the **Priority** for the candidate RP. The firewall compares the priority of the candidate RP to the priority of other candidate RPs to determine

which one acts as RP for the specified groups; the firewall selects the candidate RP with the lowest priority value (range is 0 to 255; default is 192).

- (Optional) Change the **Advertisement Interval (sec)** (range is 1 to 26,214; default is 60).
 - Enter a **Group List** of multicast groups that communicate with the RP.
 - **None**—Select if this virtual router is not an RP.
3. Add a Remote Rendezvous Point and enter the **IP Address** of that remote (external) RP.
 4. Add the multicast **Group Addresses** for which the specified remote RP address acts as RP.
 5. Select **Override learned RP for the same group** so that the external RP you configured statically serves as RP instead of an RP that is dynamically learned (elected) for the groups in the Group Addresses list.
 6. Click **OK**.

STEP 3 | Specify a group of interfaces that share a multicast configuration (IGMP, PIM, and group permissions).

1. On the **Interfaces** tab, Add a **Name** for the interface group.
2. Enter a **Description**.
3. Add an **Interface** and select one or more Layer 3 interfaces that belong to the interface group.

STEP 4 | (Optional) Configure multicast group permissions for the interface group. By default, the interface group accepts IGMP membership reports and PIM join messages from all groups.

1. Select **Group Permissions**.
2. To configure Any-Source Multicast (ASM) groups for this interface group, in the Any Source window, Add a **Name** to identify a multicast group that accepts IGMP membership reports and PIM join messages from any source.
3. Enter the multicast **Group** address or group address and /prefix that can receive multicast packets from any source on these interfaces.
4. Select **Included** to include the ASM **Group** address in the interface group (default). De-select **Included** to easily exclude an ASM group from the interface group, such as during testing.
5. Add additional multicast **Groups** (for the interface group) that want to receive multicast packets from any source.
6. To configure Source-Specific Multicast (SSM) groups in this interface group, in the Source Specific window, Add a **Name** to identify a multicast group and source address

- pair. Don't use a name that you used for Any Source multicast. (You must use IGMPv3 to configure SSM.)
7. Enter the multicast **Group** address or group address and /prefix of the group that wants to receive multicast packets from the specified source only (and can receive the packets on these interfaces).
-  A *Source Specific* group for which you specify permissions is a group that the virtual router must treat as source-specific. Configure **Source Specific Address Space** (Step 9) that includes the source-specific groups for which you configured permission.
8. Enter the **Source** IP address from which this multicast group can receive multicast packets.
 9. Select **Included** to include the SSM Group and source address pair in the interface group (default). De-select **Included** to easily exclude the pair from the interface group, such as during testing.
 10. **Add** additional multicast **Groups** (for the interface group) that receive multicast packets from a specific source only.

Virtual Router - Multicast - Interface Group

Group Permissions					IGMP	PIM		
INTERFACE		ethernet1/4						
Any Source		Source Specific						
<input type="checkbox"/>	NAME	GROUP	INCLUDED	<input type="checkbox"/>	NAME	GROUP	SOURCE	INCLUDED
<input checked="" type="checkbox"/>	video	226.4.35.9/8	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	market52	227.62.14.8	192.168.6.5	<input checked="" type="checkbox"/>
+ Add		Delete			Move Up		Move Down	
+ Add		Delete			Move Up		Move Down	
					OK		Cancel	

- STEP 5 |** Configure IGMP for the interface group if an interface faces multicast receivers, which must use IGMP to join a group.
1. On the **IGMP** tab, **Enable IGMP** (default).
 2. Specify **IGMP** parameters for interfaces in the interface group:
 - **IGMP Version—1, 2, or 3** (default).
 - **Enforce Router-Alert IP Option** (disabled by default)—Select this option if you require incoming IGMP packets that use IGMPv2 or IGMPv3 to have the **IP Router Alert Option**, RFC 2113.
 - **Robustness**—A variable that the firewall uses to tune the Group Membership Interval, Other Querier Present Interval, Startup Query Count, and Last Member Query Count

(range is 1 to 7; default is 2). Increase the value if the subnet on which this firewall is located is prone to losing packets.

- **Max Sources**—Maximum number of sources that IGMP can process simultaneously for an interface (range is 1 to 65,535; default is **unlimited**).
- **Max Groups**—Maximum number of groups that IGMP can process simultaneously for an interface (range is 1 to 65,535; default is **unlimited**).
- **Query Interval**—Number of seconds between IGMP membership Query messages that the virtual router sends to a receiver to determine whether the receiver still wants to receive the multicast packets for a group (range is 1 to 31,744; default is 125).
- **Max Query Response Time (sec)**—Maximum number of seconds allowed for a receiver to respond to an IGMP membership Query message before the virtual router determines that the receiver no longer wants to receive multicast packets for the group (range is 0 to 3,174.4; default is 10).
- **Last Member Query Interval (sec)**—Number of seconds allowed for a receiver to respond to a Group-Specific Query that the virtual router sends after a receiver sends a Leave Group message (range is 0.1 to 3,174.4; default is 1).
- **Immediate Leave** (disabled by default)—When there is only one member in a multicast group and the virtual router receives an IGMP Leave message for that group, the Immediate Leave setting causes the virtual router to remove that group and outgoing interface from the multicast routing information base (mRIB) and multicast forwarding information base (mFIB) immediately, rather than waiting for the Last Member Query Interval to expire. The Immediate Leave setting saves network resources. You cannot select Immediate Leave if the interface group uses IGMPv1.

STEP 6 | Configure PIM Sparse Mode (PIM-SM) for the interface group.

1. On the **PIM** tab, **Enable PIM** (enabled by default).
2. Specify PIM parameters for the interface group:
 - **Assert Interval**—Number of seconds between **PIM Assert messages** that the virtual router sends to other PIM routers on the multiaccess network when they are electing a PIM forwarder (range is 0 to 65,534; default is 177).
 - **Hello Interval**—Number of seconds between PIM Hello messages that the virtual router sends to its PIM neighbors from each interface in the interface group (range is 0 to 18,000; default is 30).
 - **Join Prune Interval**—Number of seconds between PIM Join messages (and between PIM Prune messages) that the virtual router sends upstream toward a multicast source (range is 1 to 18,000; default is 60).
 - **DR Priority**—Designated Router (DR) priority that controls which router in a multiaccess network forwards PIM Join and Prune messages to the RP (range is 0

to 4,294,967,295; default is 1). The DR priority takes precedence over IP address comparisons to elect the DR.

- **BSR Border**—Select this option if the interfaces in the interface group are on a virtual router that is the BSR located at the border of an enterprise LAN. This will prevent RP candidacy BSR messages from leaving the LAN.
3. **Add one or more Permitted PIM Neighbors** by specifying the **IP Address** of each router from which the virtual router accepts multicast packets.

STEP 7 | Click **OK** to save the interface group settings.

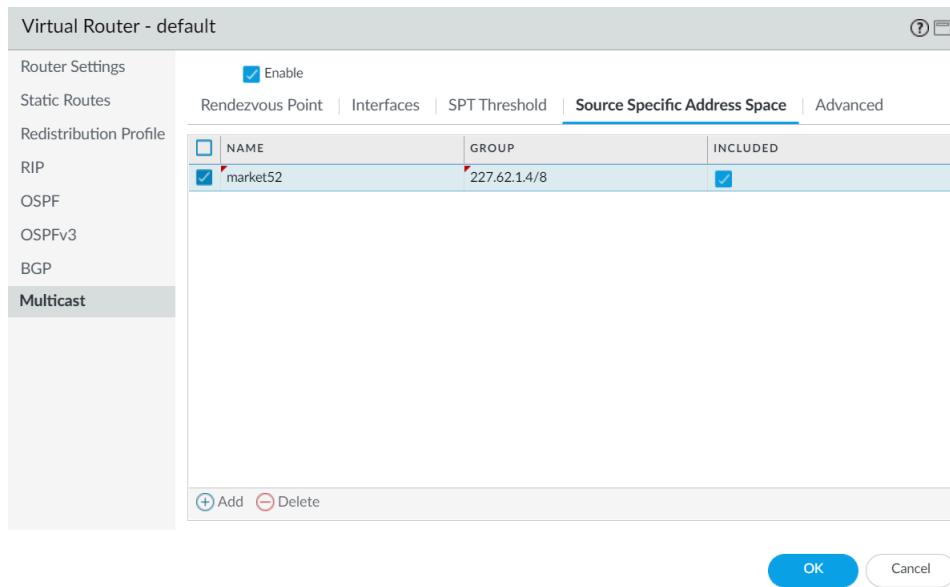
STEP 8 | (Optional) Change the Shortest-Path Tree (SPT) threshold, as described in [Shortest-Path Tree \(SPT\) and Shared Tree](#).

1. Select **SPT Threshold** and **Add a Multicast Group/Prefix**, the multicast group or prefix for which you are specifying the distribution tree.
2. Specify the **Threshold (kb)**—The point at which routing to the specified multicast group or prefix switches from shared tree (sourced from the RP) to SPT distribution:
 - **0 (switch on first data packet)** (default)—The virtual router switches from shared tree to SPT for the group or prefix when the virtual router receives the first data packet for the group or prefix.
 - **never (do not switch to spt)**—The virtual router continues to use the shared tree to forward packets to the group or prefix.
 - Enter the total number of kilobits from multicast packets that can arrive for the multicast group or prefix at any interface and over any time period, upon which the virtual router changes to SPT distribution for that multicast group or prefix.

STEP 9 | Identify the multicast groups or groups and prefixes that accept multicast packets only from a specific source.

1. Select **Source Specific Address Space** and **Add the Name** for the space.
2. Enter the multicast **Group** address with prefix length to identify the address space that receives multicast packets from a specific source. If the virtual router receives a multicast packet for an SSM group but the group is not covered by a **Source Specific Address Space**, the virtual router drops the packet.
3. Select **Included** to include the source-specific address space as a multicast group address range from which the virtual router will accept multicast packets that originated from an

- allowed specific source. De-select **Included** to easily exclude a group address space for testing.
- Add other source-specific address spaces to include all those groups for which you specified SSM group permission.



STEP 10 | (Optional) Change the length of time that a multicast route remains in the mRIB after the session ends between a multicast group and a source.

- Select the **Advanced** tab.
- Specify the **Multicast Route Age Out Time (sec)** (range is 210 to 7,200; default is 210).

STEP 11 | Click **OK** to save the multicast configuration.

STEP 12 | Create a Security policy rule to allow multicast traffic to the destination zone.

- Create a **Security Policy Rule** and on the **Destination** tab, select **multicast** or **any** for the **Destination Zone**. The **multicast** zone is a predefined Layer 3 zone that matches all multicast traffic. The **Destination Address** can be a multicast group address.
- Configure the rest of the Security policy rule.

STEP 13 | (Optional) Enable buffering of multicast packets before a route is set up.

- Select **Device > Setup > Session** and edit Session Settings.
- Enable **Multicast Route Setup Buffering** (disabled by default). The firewall can preserve the first packet(s) from a multicast flow if an entry for the corresponding multicast group does not yet exist in the multicast forwarding table (mFIB). The **Buffer Size** controls how many packets the firewall buffers from a flow. After the route is installed in the mFIB, the firewall automatically forwards the buffered first packet(s) to the receiver. (You need to enable multicast route setup buffering only if your content servers are

- directly connected to the firewall and your multicast application cannot withstand the first packet of the flow being dropped.)
3. **(Optional)** Change the **Buffer Size**. Buffer size is the number of packets per multicast flow that the firewall can buffer until the mFIB entry is set up (range is 1 to 2,000; default is 1,000). The firewall can buffer a maximum of 5,000 packets total (for all flows).
 4. Click **OK**.

STEP 14 | Commit your changes.

STEP 15 | View IP Multicast Information to view mRIB and mFIB entries, IGMP interface settings, IGMP group memberships, PIM ASM and SSM modes, group mappings to RPs, DR addresses, PIM settings, PIM neighbors, and more.

STEP 16 | If you [Configure a Static Route](#) for multicast traffic, you can install the route only in the multicast routing table (not the unicast routing table) so that the route is used for multicast traffic only.

STEP 17 | If you enable IP multicast, it is not necessary to [Configure BGP with MP-BGP for IPv4 Multicast](#) unless you have a logical multicast topology separate from a logical unicast topology. You configure MP-BGP extensions with the IPv4 address family and multicast subsequent address family only when you want to advertise multicast source prefixes into BGP under multicast subsequent address family.

View IP Multicast Information

After you [Configure IP Multicast](#) routing, view multicast routes, forwarding entries, and information about your IGMP and PIM interfaces.

- Select **Network > Virtual Routers** and in the row for the virtual router you configured, click **More Runtime Stats**.
 1. Select **Routing > Route Table** and then the **Multicast** radio button to display only multicast routes (destination IP multicast group, the next hop toward that group, and outgoing interface). This information comes from the mRIB.
 2. Select **Multicast > FIB** to view multicast route information from the mFIB: multicast groups to which the virtual router belongs, the corresponding source, incoming interfaces, and outgoing interfaces toward the receivers.

The screenshot shows the 'Virtual Router - default' configuration page. The 'Multicast' tab is selected in the top navigation bar. Below it, the 'FIB' tab is also selected. A search bar at the top right shows '2 items'. The main area displays a table with two rows of data:

GROUP	SOURCE	INCOMING INTERFACES	OUTGOING INTERFACES
226.1.1.12	160.1.1.2	ethernet1/1	tunnel.1
226.1.1.12	0.0.0.0		tunnel.1

- 3. Select **Multicast > IGMP > Interface** to view IGMP-enabled interfaces, the associated IGMP version, IP address of the IGMP Querier, Querier up time and expiry time, the

robustness setting, limits on numbers of multicast groups and sources, and whether the interface is configured for immediate leave.

Virtual Router - vr2								
Multicast								
IGMP								
Interface								
Membership								
3 items → X								
INTERFACE LEAVE	VERSION	QUERIER	QUERIER UP TIME	QUERIER EXPIRY TIME	ROBUSTNESS	GROUPS LIMIT	SOURCES LIMIT	IMMEDIATE LEAVE
ethernet1/2	3	19.19.19.1			2	0	0	no ↗
ethernet1/3	3	20.20.20.1			2	0	0	no ↗
ethernet1/8	3	192.168.5.3			2	0	0	no

4. Select **Multicast > IGMP > Membership** to see IGMP-enabled interfaces and the multicast groups to which they belong, the source, and other IGMP information.

Virtual Router - default								
Multicast								
IGMP								
Interface								
Membership								
1 item → X								
INTERFACE	GROUP	SOURCE	UP TIME	EXPIRY TIME	FILTER MODE	EXCLUDE EXPIRY	V1 HOST TIMER	V2 HOST TIMER
ethernet1/1	226.1.1.12		273.79				0.00	168.83

5. Select **Multicast > PIM > Group Mapping** to view multicast groups mapped to an RP, the origin of the RP mapping, the PIM mode for the group (ASM or SSM), and whether the

group is inactive. Groups in SSM mode don't use an RP, so the RP address displayed is 0.0.0.0. The default SSM group is 232.0.0.0/8.

The screenshot shows a table titled "Group Mapping" under the "Multicast" tab. The columns are GROUP, RP, ORIGIN, PIM MODE, and INACTIVE. There are four entries:

GROUP	RP	ORIGIN	PIM MODE	INACTIVE
224.0.55.55/32	0.0.0.0	CONFIG	SSM	no
232.0.0.0/8	0.0.0.0	CONFIG	SSM	no
238.1.1.1/32	20.20.20.10	CONFIG	ASM	no
239.255.255.250/32	20.20.20.10	CONFIG	ASM	no

- Select **Multicast > PIM > Interface** to view the IP address of the DR for an interface; the DR priority; the Hello, Join/Prune, and Assert intervals; and whether the interface is a bootstrap router (BSR).

The screenshot shows a table titled "Interface" under the "Multicast" tab. The columns are INTERFACE, ADDRESS, DR, HELLO INTERVAL, JOIN/PRUNE INTERVAL, ASSERT INTERVAL, DR PRIORITY, and BSR BORDER. There are three entries:

INTERFACE	ADDRESS	DR	HELLO INTERVAL	JOIN/PRUNE INTERVAL	ASSERT INTERVAL	DR PRIORITY	BSR BORDER
ethernet1/2	19.19.19.1	19.19.19.1	30	60	177	1	no
ethernet1/3	20.20.20.1	20.20.20.1	30	60	177	1	no
ethernet1/8	192.168.5.3	192.168.5.3	30	60	177	1	no

- Select **Multicast > PIM > Neighbor** to view information about routers that are PIM neighbors to the virtual router.

The screenshot shows a table titled "Neighbor" under the "Multicast" tab. The columns are INTERFACE, ADDRESS, SECONDARY ADDRESS, UP TIME, EXPIRY TIME, GENERATION ID, and DR PRIORITY. There is one entry:

INTERFACE	ADDRESS	SECONDARY ADDRESS	UP TIME	EXPIRY TIME	GENERATION ID	DR PRIORITY
tunnel.1	111.111.111.14		6239.49	80.22	1992867278	1

Route Redistribution

Learn about and configure route redistribution to increase accessibility of network traffic.

- [Route Redistribution Overview](#)
- [Configure Route Redistribution](#)

Route Redistribution Overview

Route redistribution on the firewall is the process of making routes that the firewall learned from one routing protocol (or a static or connected route) available to a different routing protocol, thereby increasing accessibility of network traffic. Without route redistribution, a router or virtual router advertises and shares routes only with other routers that run the same routing protocol. You can redistribute IPv4 or IPv6 BGP, connected, or static routes into the OSPF RIB and redistribute OSPFv3, connected, or static routes into the BGP RIB.

This means, for example, you can make specific networks that were once available only by manual static route configuration on specific routers available to BGP autonomous systems or OSPF areas. You can also advertise locally connected routes, such as routes to a private lab network, into BGP autonomous systems or OSPF areas.

You might want to give users on your internal OSPFv3 network access to BGP so they can access devices on the internet. In this case you would redistribute BGP routes into the OSPFv3 RIB.

Conversely, you might want to give your external users access to some parts of your internal network, so you make internal OSPFv3 networks available through BGP by redistributing OSPFv3 routes into the BGP RIB.

To [Configure Route Redistribution](#), begin by creating a redistribution profile.

Configure Route Redistribution

Perform the following procedure to configure [route redistribution](#).

STEP 1 | Create a redistribution profile.

1. Select **Network > Virtual Routers** and select a virtual router.
2. Select **Redistribution Profile** and **IPv4 or IPv6** and **Add** a profile.
3. Enter a **Name** for the profile, which must start with an alphanumeric character and can contain zero or more underscores (_), hyphens (-), dots (.), or spaces (up to 16 characters).
4. Enter a **Priority** for the profile in the range 1 to 255. The firewall matches routes to profiles in order using the profile with the highest priority (lowest priority value) first. Higher priority rules take precedence over lower priority rules.
5. For **Redistribute**, select one of the following:
 - **Redist**—Select for redistribution the routes that match this filter.
 - **No Redist**—Select for redistribution routes that match the redistribution profiles except the routes that match this filter. This selection treats the profile as a block list that specifies which routes not to select for redistribution. For example, if you have multiple redistribution profiles for BGP, you can create a **No Redist** profile to exclude several prefixes, and then a general redistribution profile with a lower priority (higher priority value) after it. The two profiles combine and the higher priority profile takes precedence. You can't have only **No Redist** profiles; you would always need at least one **Redist** profile to redistribute routes.
6. On the **General Filter** tab, for **Source Type**, select one or more types of route to redistribute:
 - **bgp**—Redistribute BGP routes that match the profile.
 - **connect**—Redistribute connected routes that match the profile.
 - **ospf (IPv4 only)**—Redistribute OSPF routes that match the profile.
 - **rip (IPv4 only)**—Redistribute RIP routes that match the profile.
 - **ospfv3 (IPv6 only)**—Redistribute OSPFv3 routes that match the profile.
 - **static**—Redistribute static routes that match the profile.
7. (**Optional**) For **Interface**, **Add** one or more egress interfaces of associated routes to match for redistribution. To remove an entry, click **Delete**.
8. (**Optional**) For **Destination**, **Add** one or more IPv4 or IPv6 destinations of routes to match for redistribution. To remove an entry, click **Delete**.
9. (**Optional**) For **Next Hop**, **Add** one or more next hop IPv4 or IPv6 addresses of routes to match for redistribution. To remove an entry, click **Delete**.
10. Click **OK**.

STEP 2 | (Optional—When General Filter includes ospf or ospfv3) Create an OSPF filter to further specify which OSPF or OSPFv3 routes to redistribute.

1. Select **Network > Virtual Routers** and select the virtual router.
2. Select **Redistribution Profile** and **IPv4** or **IPv6** and select the profile you created.
3. Select **OSPF Filter**.
4. For Path Type, select one or more of the following types of OSPF path to redistribute: **ext-1**, **ext-2**, **inter-area**, or **intra-area**.
5. To specify an **Area** from which to redistribute OSPF or OSPFv3 routes, **Add** an area in IP address format.
6. To specify a **Tag**, **Add** a tag in IP address format.
7. Click **OK**.

STEP 3 | (Optional—When General Filter includes bgp) Create a BGP filter to further specify which BGP routes to redistribute.

1. Select **Network > Virtual Routers** and select the virtual router.
2. Select **Redistribution Profile** and **IPv4** or **IPv6** and select the profile you created.
3. Select **BGP Filter**.
4. For **Community**, **Add** to select from the list of communities, such as well-known communities: **local-as**, **no-advertise**, **no-export**, or **nopeer**. You can also enter a 32-bit value in decimal or hexadecimal or in AS:VAL format, where AS and VAL are each in the range 0 to 65,535. Enter a maximum of 10 entries.
5. For **Extended Community**, **Add** an extended community as a 64-bit value in hexadecimal or in TYPE:AS:VAL or TYPE:IP:VAL format. TYPE is 16 bits; AS or IP is 16 bits; VAL is 32 bits. Enter a maximum of five entries.
6. Click **OK**.

STEP 4 | Select the protocol into which you are redistributing routes, and set the attributes for those routes.

This task illustrates redistributing routes into BGP.

1. Select **Network > Virtual Routers** and select the virtual router.
2. Select **BGP > Redist Rules**.
3. Select **Allow Redistribute Default Route** to allow the firewall to redistribute the default route.
4. Click **Add**.
5. Select **Address Family Type: IPv4 or IPv6** to specify in which route table the redistributed routes will be put.
6. Select the **Name** of the Redistribution profile you created, which selects the routes to redistribute.
7. **Enable** the redistribution rule.
8. (**Optional**) Enter any of the following values, which the firewall applies to the routes being redistributed:
 - **Metric** in the range 1 to 65,535.
 - **Set Origin**—Origin of the route: **igp**, **egp**, or **incomplete**.
 - **Set MED**—MED value in the range 0 to 4,294,967,295.
 - **Set Local Preference**—Local preference value in the range 0 to 4,294,967,295.
 - **Set AS Path Limit**—Maximum number of autonomous systems in the **AS_PATH** in the range 1 to 255.
 - **Set Community**—Select or enter a 32-bit value in decimal or hexadecimal, or enter a value in **AS:VAL** format, where **AS** and **VAL** are each in the range 0 to 65,525. Enter a maximum of 10 entries.
 - **Set Extended Community**—Select or enter an extended community as a 64-bit value in hexadecimal or in **TYPE:AS:VAL** or **TYPE:IP:VAL** format. **TYPE** is 16 bits; **AS** or **IP** is 16 bits; **VAL** is 32 bits. Enter a maximum of five entries.
9. Click **OK**.

STEP 5 | Commit your changes.

GRE Tunnels

The Generic Routing Encapsulation (GRE) tunnel protocol is a carrier protocol that encapsulates a payload protocol. The GRE packet itself is encapsulated in a transport protocol (IPv4 or IPv6).

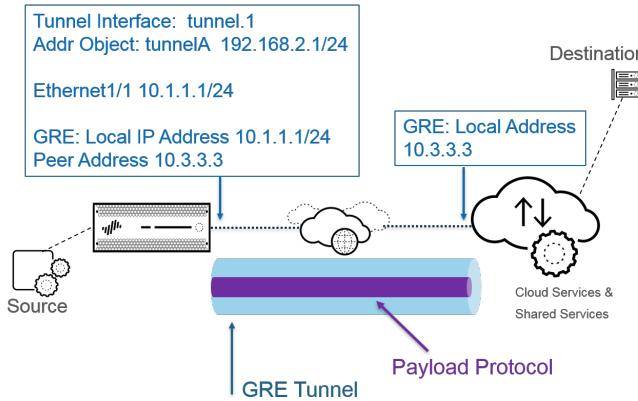
- [GRE Tunnel Overview](#)
- [Create a GRE Tunnel](#)

GRE Tunnel Overview

A Generic Routing Encapsulation (GRE) tunnel connects two endpoints (a firewall and another appliance) in a point-to-point, logical link. The firewall can terminate GRE tunnels; you can route or forward packets to a GRE tunnel. GRE tunnels are simple to use and often the tunneling protocol of choice for point-to-point connectivity, especially to services in the cloud or to partner networks.

[Create a GRE tunnel](#) when you want to direct packets that are destined for an IP address to take a certain point-to-point path, for example to a cloud-based proxy or to a partner network. The packets travel through the GRE tunnel (over a transit network such as the internet) to the cloud service while on their way to the destination address. This enables the cloud service to enforce its services or policies on the packets.

The following figure is an example of a GRE tunnel connecting the firewall across the internet to a cloud service.



For better performance and to avoid single points of failure, split multiple connections to the firewall among multiple GRE tunnels rather than use a single tunnel. Each GRE tunnel needs a tunnel interface.

When the firewall allows a packet to pass (based on a policy match) and the packet egresses to a GRE tunnel interface, the firewall adds GRE encapsulation; it doesn't generate a session. The firewall does not perform a Security policy rule lookup for the GRE-encapsulated traffic, so you don't need a Security policy rule for the GRE traffic that the firewall encapsulates. However, when the firewall receives GRE traffic, it generates a session and applies all policies to the GRE IP header in addition to the encapsulated traffic. The firewall treats the received GRE packet like any other packet. Therefore:

- If the firewall receives the GRE packet on an interface that has the same zone as the tunnel interface associated with the GRE tunnel (for example, tunnel.1), the source zone is the same as the destination zone. By default, traffic is allowed within a zone (intrazone traffic), so the ingress GRE traffic is allowed by default.
- However, if you configured your own intrazone Security policy rule to deny such traffic, you must explicitly allow GRE traffic.

- Likewise, if the zone of the tunnel interface associated with the GRE tunnel (for example, tunnel.1) is a different zone from that of the ingress interface, you must configure a Security policy rule to allow the GRE traffic.

Because the firewall encapsulates the tunneled packet in a GRE packet, the additional 24 bytes of GRE header automatically result in a smaller [Maximum Segment Size \(MSS\)](#) in the maximum transmission unit (MTU). If you don't change the IPv4 MSS Adjustment Size for the interface, the firewall reduces the MTU by 64 bytes by default (40 bytes of IP header + 24 bytes of GRE header). This means if the default MTU is 1,500 bytes, the MSS will be 1,436 bytes ($1,500 - 40 - 24 = 1,436$). If you configure an MSS Adjustment Size of 300 bytes, for example, the MSS will be only 1,176 bytes ($1,500 - 300 - 24 = 1,176$).

The firewall does not support routing a GRE or IPSec tunnel to a GRE tunnel, but you can route a GRE tunnel to an IPSec tunnel. Additionally:

- A GRE tunnel does not support QoS.
- The firewall does not support a single interface acting as both a GRE tunnel endpoint and a decryption broker.
- GRE tunneling does not support NAT between GRE tunnel endpoints.



If you need to connect to another vendor's network, we recommend you [set up an IPSec tunnel](#), not a GRE tunnel; you should use a GRE tunnel only if that is the only point-to-point tunnel mechanism that the vendor supports. You can also enable GRE over IPSec if the remote endpoint requires that ([Add GRE Encapsulation](#)). Add GRE encapsulation in cases where the remote endpoint requires traffic to be encapsulated within a GRE tunnel before IPSec encrypts the traffic. For example, some implementations require multicast traffic to be encapsulated before IPSec encrypts it. If this is a requirement for your environment and the GRE tunnel and IPSec tunnel share the same IP address, [Add GRE Encapsulation](#) when you set up the IPSec tunnel.



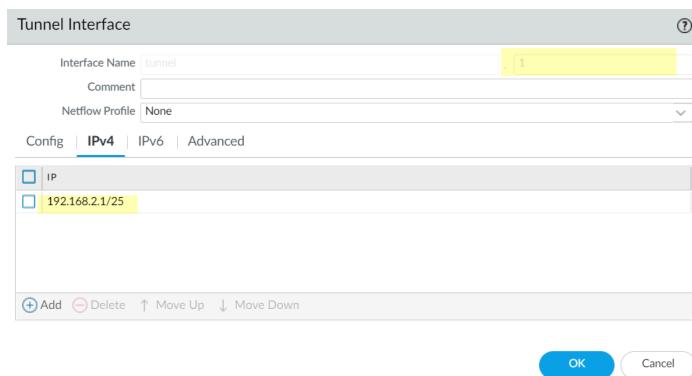
If you aren't planning to terminate a GRE tunnel on the firewall, but you want the ability to inspect and control traffic passing through the firewall inside a GRE tunnel, don't create a GRE tunnel. Instead, perform [Tunnel Content Inspection](#) of GRE traffic. With tunnel content inspection, you are inspecting and enforcing policy on GRE traffic passing through the firewall, not creating a point-to-point, logical link for the purpose of directing traffic.

Create a GRE Tunnel

Create a **Generic Routing Encapsulation (GRE)** tunnel to connect two endpoints in a point-to-point, logical link.

STEP 1 | Create a tunnel interface.

1. Select **Network > Interfaces > Tunnel**.
2. **Add** a tunnel and enter the tunnel **Interface Name** followed by a period and a number (range is 1 to 9,999). For example, **tunnel.1**.
3. On the **Config** tab, assign the tunnel interface to a **Virtual Router**.
4. Assign the tunnel interface to a **Virtual System** if the firewall supports multiple virtual systems.
5. Assign the tunnel interface to a **Security Zone**.



6. Assign an IP address to the tunnel interface. (You must assign an IP address if you want to route to this tunnel or monitor the tunnel endpoint.) Select **IPv4** or **IPv6** or configure both.



This address and the corresponding address of the tunnel interface of the peer should be on the same subnet because it is a point-to-point, logical link.

- (**IPv4 only**) On the **IPv4** tab, **Add** an IPv4 address, select an address object, or click **New Address** and specify the **Type** of address and enter it. For example, enter **192.168.2.1**.
 - (**IPv6 only**) On the **IPv6** tab, **Enable IPv6 on the interface**.
 1. For **Interface ID**, select **EUI-64 (default 64-bit Extended Unique Identifier)**.
 2. **Add** a new **Address**, select an IPv6 address object, or click **New Address** and specify an address **Name**. **Enable address on interface** and click **OK**.
 3. Select **Type** of address and enter the IPv6 address or FQDN and click **OK** to save the new address.
 4. Select **Enable address on interface** and click **OK**.
7. Click **OK**.

STEP 2 | Create a GRE tunnel to force packets to traverse a specific point-to-point path.

1. Select **Network > GRE Tunnels** and **Add** a tunnel by **Name**.
2. Select the **Interface** to use as the local GRE tunnel endpoint (source interface), which is an Ethernet interface or subinterface, an Aggregate Ethernet (AE) interface, a loopback interface, or a VLAN interface.
3. Select the **Local Address** to be **IP** and select the IP address of the interface you just selected.
4. Enter the **Peer Address**, which is the IP address of the opposite endpoint of the GRE tunnel.
5. Select the **Tunnel Interface** that you created in Step 1. (This identifies the tunnel when it is the egress **Interface** for routing.)
6. Enter the **TTL** for the IP packet encapsulated in the GRE packet (range is 1 to 255; default is 64).
7. Select **Copy ToS Header** to copy the Type of Service (ToS) field from the inner IP header to the outer IP header of the encapsulated packets to preserve the original ToS information. Select this option if your network uses QoS and depends on the ToS bits for enforcing QoS policies.

**STEP 3 | (Best Practice)** Enable the Keep Alive function for the GRE tunnel.

 If Keep Alive is enabled, by default it takes three unreturned keepalive packets (Retries) at 10-second intervals for the GRE tunnel to go down and it takes five Hold Timer intervals at 10-second intervals for the GRE tunnel to come back up.

1. Select **Keep Alive** to enable the keepalive function for the GRE tunnel (default is disabled).
2. (**Optional**) Set the **Interval (sec)** (in seconds) between keepalive packets that the local end of the GRE tunnel sends to the tunnel peer. This is also the interval that, when multiplied by the **Hold Timer**, is the length of time that the firewall must see successful keepalive packets before the GRE tunnel comes back up (range is 1 to 50; default is 10). Setting an interval too small will cause many keepalive packets that might be unnecessary in your environment and will require extra bandwidth and processing. Setting an interval too large can delay failover because error conditions might not be identified immediately.
3. (**Optional**) Enter the **Retry** setting, which is the number of intervals that keepalive packets are not returned before the firewall considers the tunnel peer down (range is 1 to 5).

- to 255; default is 3). When the tunnel is down, the firewall removes routes associated with the tunnel from the forwarding table. Configuring a retry setting helps avoid taking measures on a tunnel that is not really down.
4. (**Optional**) Set the **Hold Timer**, which is the number of **Intervals** that keepalive packets are successful, after which the firewall re-establishes communication with the tunnel peer (range is 1 to 64; default is 5).

STEP 4 | Click **OK**.

STEP 5 | Configure a routing protocol or static route to route traffic to the destination by way of the GRE tunnel. For example, [Configure a Static Route](#) to the network of the destination server and specify the egress **Interface** to be the local tunnel endpoint (tunnel.1). Configure the Next Hop to be the IP address of the tunnel at the opposite end. For example, 192.168.2.3.

STEP 6 | Commit your changes.

STEP 7 | Configure the opposite end of the tunnel with its public IP address, its local and peer IP addresses (that correspond to the peer and local IP addresses, respectively, of the GRE tunnel on the firewall), and its routing protocol or static route.

STEP 8 | Verify that the firewall can communicate with the tunnel peer over the GRE tunnel.

1. [Access the CLI](#).
2. > **ping source 192.168.2.1 host 192.168.2.3**

DHCP

This section describes Dynamic Host Configuration Protocol (DHCP) and the tasks required to configure an interface on a Palo Alto Networks® firewall to act as a DHCP server, client, or relay agent. By assigning these roles to different interfaces, the firewall can perform multiple roles.

- [DHCP Overview](#)
- [Firewall as a DHCP Server and Client](#)
- [DHCP Messages](#)
- [DHCP Addressing](#)
- [DHCP Options](#)
- [Configure an Interface as a DHCP Server](#)
- [Configure an Interface as a DHCP Client](#)
- [Configure the Management Interface as a DHCP Client](#)
- [Configure an Interface as a DHCP Relay Agent](#)
- [Monitor and Troubleshoot DHCP](#)

DHCP Overview

DHCP is a standardized protocol defined in [RFC 2131, Dynamic Host Configuration Protocol](#).

DHCP has two main purposes: to provide TCP/IP and link-layer configuration parameters and to provide network addresses to dynamically configured hosts on a TCP/IP network.

DHCP uses a client-server model of communication. This model consists of three roles that the device can fulfill: DHCP client, DHCP server, and DHCP relay agent.

- A device acting as a DHCP client (host) can request an IP address and other configuration settings from a DHCP server. Users on client devices save configuration time and effort, and need not know the network's addressing plan or other resources and options they are inheriting from the DHCP server.
- A device acting as a DHCP server can service clients. By using any of three [DHCP Addressing](#) mechanisms, the network administrator saves configuration time and has the benefit of reusing a limited number of IP addresses when a client no longer needs network connectivity. The server can deliver IP addressing and many DHCP options to many clients.
- A device acting as a DHCP relay agent transmits DHCP messages between DHCP clients and servers.

DHCP uses [User Datagram Protocol \(UDP\)](#), [RFC 768](#), as its transport protocol. DHCP messages that a client sends to a server are sent to well-known port 67 (UDP—Bootstrap Protocol and DHCP). [DHCP Messages](#) that a server sends to a client are sent to port 68.

An interface on a Palo Alto Networks[®] firewall can perform the role of a DHCP server, client, or relay agent. The interface of a DHCP server or relay agent must be a Layer 3 Ethernet, Aggregated Ethernet, or Layer 3 VLAN interface. You configure the firewall interfaces with the appropriate settings for any combination of roles. The behavior of each role is summarized in [Firewall as a DHCP Server and Client](#).

The firewall supports DHCPv4 Server and DHCPv6 Relay.

The Palo Alto Networks implementations of DHCP server and DHCP client support IPv4 addresses only. Its DHCP relay implementation supports IPv4 and IPv6. DHCP client is not supported in High Availability active/active mode.

Firewall as a DHCP Server and Client

The firewall can function as a DHCP server and as a DHCP client. [Dynamic Host Configuration Protocol, RFC 2131](#), is designed to support IPv4 and IPv6 addresses. The Palo Alto Networks® implementation of DHCP server supports IPv4 addresses only.

The firewall DHCP server operates in the following manner:

- When the DHCP server receives a DHCPDISCOVER message from a client, the server replies with a DHCPOFFER message containing all of the predefined and user-defined options in the order they appear in the configuration. The client selects the options it needs and responds with a DHCPREQUEST message.
- When the server receives a DHCPREQUEST message from a client, the server replies with its DHCPACK message containing only the options specified in the request.

The firewall DHCP client operates in the following manner:

- When the DHCP client receives a DHCPOFFER from the server, the client automatically caches all of the options offered for future use, regardless of which options it had sent in its DHCPREQUEST.
- By default and to save memory consumption, the client caches only the first value of each option code if it receives multiple values for a code.
- There is no maximum length for DHCP messages unless the DHCP client specifies a maximum in option 57 in its DHCPDISCOVER or DHCPREQUEST messages.

DHCP Messages

DHCP uses eight standard message types, which are identified by an option type number in the DHCP message. For example, when a client wants to find a DHCP server, it broadcasts a DHCPDISCOVER message on its local physical subnet. If there is no DHCP server on its subnet and if DHCP Helper or DHCP Relay is configured properly, the message is forwarded to DHCP servers on a different physical subnet. Otherwise, the message will go no further than the subnet on which it originated. One or more DHCP servers will respond with a DHCPOFFER message that contains an available network address and other configuration parameters.

When the client needs an IP address, it sends a DHCPREQUEST to one or more servers. Of course if the client is requesting an IP address, it doesn't have one yet, so [RFC 2131](#) requires that the broadcast message the client sends out have a source address of 0 in its IP header.

When a client requests configuration parameters from a server, it might receive responses from more than one server. Once a client has received its IP address, it is said that the client has at least an IP address and possibly other configuration parameters *bound* to it. DHCP servers manage such binding of configuration parameters to clients.

The following table lists the DHCP messages.

DHCP Message	Description
DHCPDISCOVER	Client broadcast to find available DHCP servers.
DHCPOFFER	Server response to client's DHCPDISCOVER, offering configuration parameters.
DHCPREQUEST	Client message to one or more servers to do any of the following: <ul style="list-style-type: none">Request parameters from one server and implicitly decline offers from other servers.Confirm that a previously allocated address is correct after, for example, a system reboot.Extend the lease of a network address.
DHCPACK	Server to client acknowledgment message containing configuration parameters, including a confirmed network address.
DHCPNAK	Server to client negative acknowledgment indicating the client's understanding of the network address is incorrect (for example, if the client has moved to a new subnet), or a client's lease has expired.
DHCPDECLINE	Client to server message indicating the network address is already being used.

DHCP Message	Description
DHCPRELEASE	Client to server message giving up the user of the network address and canceling the remaining time on the lease.
DHCPIINFORM	Client to server message requesting only local configuration parameters; client has an externally configured network address.

DHCP Addressing

- [DHCP Address Allocation Methods](#)
- [DHCP Leases](#)

DHCP Address Allocation Methods

There are three ways that a DHCP server either assigns or sends an IP address to a client:

- **Automatic allocation**—The DHCP server assigns a permanent IP address to a client from its **IP Pools**. On the firewall, a **Lease** specified as **Unlimited** means the allocation is permanent.
- **Dynamic allocation**—The DHCP server assigns a reusable IP address from **IP Pools** of addresses to a client for a maximum period of time, known as a *lease*. This method of address allocation is useful when the customer has a limited number of IP addresses; they can be assigned to clients who need only temporary access to the network. See the [DHCP Leases](#) section.
- **Static allocation**—The network administrator chooses the IP address to assign to the client and the DHCP server sends it to the client. A static DHCP allocation is permanent; it is done by configuring a DHCP server and choosing a **Reserved Address** to correspond to the **MAC Address** of the client device. The DHCP assignment remains in place even if the client logs off, reboots, has a power outage, etc.

Static allocation of an IP address is useful, for example, if you have a printer on a LAN and you do not want its IP address to keep changing, because it is associated with a printer name through DNS. Another example is if a client device is used for something crucial and must keep the same IP address, even if the device is turned off, unplugged, rebooted, or a power outage occurs, etc.

Keep these points in mind when configuring a **Reserved Address**:

- It is an address from the **IP Pools**. You may configure multiple reserved addresses.
- If you configure no **Reserved Address**, the clients of the server will receive new DHCP assignments from the pool when their leases expire or if they reboot, etc. (unless you specified that a **Lease** is **Unlimited**).
- If you allocate all of the addresses in the **IP Pools** as a **Reserved Address**, there are no dynamic addresses free to assign to the next DHCP client requesting an address.
- You may configure a **Reserved Address** without configuring a **MAC Address**. In this case, the DHCP server will not assign the **Reserved Address** to any device. You might reserve a few addresses from the pool and statically assign them to a fax and printer, for example, without using DHCP.

DHCP Leases

A lease is defined as the time period for which a DHCP server allocates a network address to a client. The lease might be extended (renewed) upon subsequent requests. If the client no longer needs the address, it can release the address back to the server before the lease is up. The server is then free to assign that address to a different client if it has run out of unassigned addresses.

The lease period configured for a DHCP server applies to all of the addresses that a single DHCP server (interface) dynamically assigns to its clients. That is, all of that interface's addresses assigned dynamically are of **Unlimited** duration or have the same **Timeout** value. A different DHCP server configured on the firewall may have a different lease term for its clients. A **Reserved Address** is a static address allocation and is not subject to the lease terms.

Per the DHCP standard, [RFC 2131](#), a DHCP client does not wait for its lease to expire, because it risks getting a new address assigned to it. Instead, when a DHCP client reaches the halfway point of its lease period, it attempts to extend its lease so that it retains the same IP address. Thus, the lease duration is like a sliding window.

Typically if an IP address was assigned to a device, the device was subsequently taken off the network and its lease was not extended, the DHCP server will let that lease run out. Because the client is gone from the network and no longer needs the address, the lease duration in the server is reached and the lease is in "Expired" state.

The firewall has a hold timer that prevents the expired IP address from being reassigned immediately. This behavior temporarily reserves the address for the device in case it comes back onto the network. But if the address pool runs out of addresses, the server re-allocates this expired address before the hold timer expires. Expired addresses are cleared automatically as the system needs more addresses or when the hold timer releases them.

In the CLI, use the **show dhcp server lease** operational command to view lease information about the allocated IP addresses. If you don't want to wait for expired leases to be released automatically, you can use the **clear dhcp lease interface <interface> expired-only** command to clear expired leases, making those addresses available in the pool again. You can use the **clear dhcp lease interface <interface> ip <ip_address>** command to release a particular IP address. Use the **clear dhcp lease interface <interface> mac <mac_address>** command to release a particular MAC address.

DHCP Options

The history of DHCP and DHCP options traces back to the Bootstrap Protocol (BOOTP). BOOTP was used by a host to configure itself dynamically during its booting procedure. A host could receive an IP address and a file from which to download a boot program from a server, along with the server's address and the address of an Internet gateway.

Included in the BOOTP packet was a vendor information field, which could contain a number of tagged fields containing various types of information, such as the subnet mask, the BOOTP file size, and many other values. [RFC 1497](#) describes the [BOOTP Vendor Information Extensions](#). DHCP replaces BOOTP; BOOTP is not supported on the firewall.

These extensions eventually expanded with the use of DHCP and DHCP host configuration parameters, also known as options. Similar to vendor extensions, DHCP options are tagged data items that provide information to a DHCP client. The options are sent in a variable-length field at the end of a DHCP message. For example, the DHCP Message Type is option 53, and a value of 1 indicates the DHCPDISCOVER message. DHCP options are defined in [RFC 2132](#), [DHCP Options and BOOTP Vendor Extensions](#).

A DHCP client can negotiate with the server, limiting the server to send only those options that the client requests.

- [Predefined DHCP Options](#)
- [Multiple Values for a DHCP Option](#)
- [DHCP Options 43, 55, and 60 and Other Customized Options](#)

Predefined DHCP Options

Palo Alto Networks® firewalls support user-defined and predefined DHCP options in the DHCP server implementation. Such options are configured on the DHCP server and sent to the clients that sent a DHCPREQUEST to the server. The clients are said to *inherit* and implement the options that they are programmed to accept.

The firewall supports the following predefined options on its DHCP servers, shown in the order in which they appear on the **DHCP Server** configuration screen:

DHCP Option	DHCP Option Name
51	Lease duration
3	Gateway
1	IP Pool Subnet (mask)
6	Domain Name System (DNS) server address (primary and secondary)
44	Windows Internet Name Service (WINS) server address (primary and secondary)

DHCP Option	DHCP Option Name
41	Network Information Service (NIS) server address (primary and secondary)
42	Network Time Protocol (NTP) server address (primary and secondary)
70	Post Office Protocol Version 3 (POP3) server address
69	Simple Mail Transfer Protocol (SMTP) server address
15	DNS suffix

As mentioned, you can also configure vendor-specific and customized options, which support a wide variety of office equipment, such as IP phones and wireless infrastructure devices. Each option code supports multiple values, which can be IP address, ASCII, or hexadecimal format. With the firewall enhanced DHCP option support, branch offices do not need to purchase and manage their own DHCP servers in order to provide vendor-specific and customized options to DHCP clients.

Multiple Values for a DHCP Option

You can enter multiple option values for an **Option Code** with the same **Option Name**, but all values for a particular code and name combination must be the same type (IP address, ASCII, or hexadecimal). If one type is inherited or entered, and later a different type is entered for the same code and name combination, the second type will overwrite the first type.

You can enter an **Option Code** more than once by using a different **Option Name**. In this case, the **Option Type** for the Option Code can differ among the multiple option names. For example, if option Coastal Server (option code 6) is configured with IP address type, option Server XYZ (option code 6) with ASCII type is also allowed.

The firewall sends multiple values for an option (strung together) to a client in order from top to bottom. Therefore, when entering multiple values for an option, enter the values in the order of preference, or else move the options to achieve your preferred order in the list. The order of options in the firewall configuration determines the order that the options appear in DHCPOFFER and DHCPACK messages.

You can enter an option code that already exists as a predefined option code, and the customized option code will override the predefined DHCP option; the firewall issues a warning.

DHCP Options 43, 55, and 60 and Other Customized Options

The following table describes the option behavior for several options described in [RFC 2132](#).

Option Code	Option Name	Option Description/Behavior
43	Vendor Specific Information	<p>Sent from server to client. Vendor-specific information that the DHCP server has been configured to offer to the client. The information is sent to the client only if the server has a Vendor Class Identifier (VCI) in its table that matches the VCI in the client's DHCPREQUEST.</p> <p>An Option 43 packet can contain multiple vendor-specific pieces of information. It can also include encapsulated, vendor-specific extensions of data.</p>
55	Parameter Request List	<p>Sent from client to server. List of configuration parameters (option codes) that a DHCP client is requesting, possibly in order of the client's preference. The server tries to respond with options in the same order.</p>
60	Vendor Class Identifier (VCI)	<p>Sent from client to server. Vendor type and configuration of a DHCP client. The DHCP client sends option code 60 in a DHCPREQUEST to the DHCP server. When the server receives option 60, it sees the VCI, finds the matching VCI in its own table, and then it returns option 43 with the value (that corresponds to the VCI), thereby relaying vendor-specific information to the correct client. Both the client and server have knowledge of the VCI.</p>

You can send custom, vendor-specific option codes that are not defined in RFC 2132. The option codes can be in the range 1-254 and of fixed or variable length.



Custom DHCP options are not validated by the DHCP Server; you must ensure that you enter correct values for the options you create.

For ASCII and hexadecimal DHCP option types, the option value can be a maximum of 255 octets.

Configure an Interface as a DHCP Server

The prerequisites for this task are:

- Configure a Layer 3 Ethernet or Layer 3 VLAN interface.
- Assign the interface to a virtual router and a zone.
- Determine a valid pool of IP addresses from your network plan that you can designate to be assigned by your DHCP server to clients.
- Collect the DHCP options, values, and Vendor Class Identifiers you plan to configure.

Capacities are as follows:

- For firewall models other than PA-5200 Series and PA-7000 Series firewalls, see the [Product Selection tool](#).
- On PA-5220 firewalls, you can configure a maximum of 500 DHCP servers and a maximum of 2,048 DHCP relay agents minus the number of DHCP servers configured. For example, if you configure 500 DHCP servers, you can configure 1,548 DHCP relay agents.
- On PA-5250, PA-5260, and PA-7000 Series firewalls, you can configure a maximum of 500 DHCP servers, and a maximum of 4,096 DHCP relay agents minus the number of DHCP servers configured. For example, if you configure 500 DHCP servers, you can configure 3,596 DHCP relay agents.

Perform the following task to configure an interface on the firewall to act as a DHCP server.

STEP 1 | Select an interface to be a DHCP Server.

1. Select **Network > DHCP > DHCP Server** and **Add an Interface** name or select one.
2. For **Mode**, select **enabled** or **auto** mode. Auto mode enables the server and disables it if another DHCP server is detected on the network. The **disabled** setting disables the server.
3. (**Optional**) Select **Ping IP when allocating new IP** if you want the server to ping the IP address before it assigns that address to its client.



If the ping receives a response, that means a different device already has that address, so it is not available. The server assigns the next address from the pool instead. This behavior is similar to [Optimistic Duplicate Address Detection \(DAD\) for IPv6, RFC 4429](#).



After you set options and return to the DHCP server tab, the **Probe IP** column for the interface indicates if **Ping IP when allocating new IP** was selected.

STEP 2 | Configure the predefined **DHCP Options** that the server sends to its clients.

- In the Options section, select a **Lease** type:
- **Unlimited** causes the server to dynamically choose IP addresses from the **IP Pools** and assign them permanently to clients.
- **Timeout** determines how long the lease will last. Enter the number of **Days** and **Hours**, and optionally the number of **Minutes**.
- **Inheritance Source**—Leave **None** or select a source DHCP client interface or PPPoE client interface to propagate various server settings into the DHCP server. If you specify an **Inheritance Source**, select one or more options below that you want **inherited** from this source.

Specifying an inheritance source allows the firewall to quickly add DHCP options from the upstream server received by the DHCP client. It also keeps the client options updated if the source changes an option. For example, if the source replaces its NTP server (which had been identified as the **Primary NTP** server), the client will automatically inherit the new address as its **Primary NTP** server.



When inheriting DHCP option(s) that contain multiple IP addresses, the firewall uses only the first IP address contained in the option to conserve cache memory. If you require multiple IP addresses for a single option, configure the DHCP options directly on that firewall rather than configure inheritance.

- **Check inheritance source status**—If you selected an **Inheritance Source**, clicking this link opens the **Dynamic IP Interface Status** window, which displays the options that were inherited from the DHCP client.
- **Gateway**—IP address of the network gateway (an interface on the firewall) that is used to reach any device not on the same LAN as this DHCP server.
- **Subnet Mask**—Network mask used with the addresses in the **IP Pools**.

For the following fields, click the down arrow and select **None**, or **inherited**, or enter a remote server's IP address that your DHCP server will send to clients for accessing that service. If you select **inherited**, the DHCP server inherits the values from the source DHCP client specified as the **Inheritance Source**.

- **Primary DNS, Secondary DNS**—IP address of the preferred and alternate Domain Name System (DNS) servers.
- **Primary WINS, Secondary WINS**—IP address of the preferred and alternate Windows Internet Naming Service (WINS) servers.
- **Primary NIS, Secondary NIS**—IP address of the preferred and alternate Network Information Service (NIS) servers.
- **Primary NTP, Secondary NTP**—IP address of the available Network Time Protocol servers.
- **POP3 Server**—IP address of Post Office Protocol (POP3) server.
- **SMTP Server**—IP address of a Simple Mail Transfer Protocol (SMTP) server.
- **DNS Suffix**—Suffix for the client to use locally when an unqualified hostname is entered that it cannot resolve.

STEP 3 | **(Optional)** Configure a vendor-specific or custom DHCP option that the DHCP server sends to its clients.

1. In the Custom DHCP Options section, **Add** a descriptive **Name** to identify the DHCP option.
2. Enter the **Option Code** you want to configure the server to offer (range is 1-254). (See [RFC 2132](#) for option codes.)
3. If the **Option Code** is 43, the **Vendor Class Identifier** field appears. Enter a VCI, which is a string or hexadecimal value (with 0x prefix) used as a match against a value that comes from the client Request containing option 60. The server looks up the incoming VCI in its table, finds it, and returns Option 43 and the corresponding option value.
4. **Inherit from DHCP server inheritance source**—Select it only if you specified an **Inheritance Source** for the DHCP Server predefined options and you want the vendor-specific and custom options also to be **inherited** from this source.
5. **Check inheritance source status**—If you selected an **Inheritance Source**, clicking this link opens **Dynamic IP Interface Status**, which displays the options that were inherited from the DHCP client.
6. If you did not select **Inherit from DHCP server inheritance source**, select an **Option Type**: **IP Address**, **ASCII**, or **Hexadecimal**. Hexadecimal values must start with the 0x prefix.
7. Enter the **Option Value** you want the DHCP server to offer for that **Option Code**. You can enter multiple values on separate lines.
8. Click **OK**.

STEP 4 | **(Optional)** Add another vendor-specific or custom DHCP option.

1. Repeat the prior step to enter another custom DHCP Option.
 - You can enter multiple option values for an **Option Code** with the same **Option Name**, but all values for an **Option Code** must be the same type (**IP Address**, **ASCII**, or **Hexadecimal**). If one type is inherited or entered and a different type is entered for the same **Option Code** and the same **Option Name**, the second type will overwrite the first type.

When entering multiple values for an option, enter the values in the order of preference, or else move the Custom DHCP Options to achieve the preferred order in the list. Select an option and click **Move Up** or **Move Down**.

 - You can enter an **Option Code** more than once by using a different **Option Name**. In this case, the **Option Type** for the Option Code can differ among the multiple option names.
2. Click **OK**.

STEP 5 | Identify the stateful pool of IP addresses from which the DHCP server chooses an address and assigns it to a DHCP client.

 If you are not the network administrator for your network, ask the network administrator for a valid pool of IP addresses from the network plan that can be designated to be assigned by your DHCP server.

1. In the **IP Pools** field, **Add** the range of IP addresses from which this server assigns an address to a client. Enter an IP subnet and subnet mask (for example, 192.168.1.0/24) or a range of IP addresses (for example, 192.168.1.10-192.168.1.20).
 - An IP Pool or a **Reserved Address** is mandatory for dynamic IP address assignment.
 - An IP Pool is optional for static IP address assignment as long as the static IP addresses that you assign fall into the subnet that the firewall interface services.
2. **(Optional)** Repeat this step to specify another IP address pool.

STEP 6 | **(Optional)** Specify an IP address from the IP pools that will not be assigned dynamically. If you also specify a **MAC Address**, the **Reserved Address** is assigned to that device when the device requests an IP address through DHCP.

 See the [DHCP Addressing section](#) for an explanation of allocation of a **Reserved Address**.

1. In the **Reserved Address** field, click **Add**.
2. Enter an IP address from the **IP Pools** (format x.x.x.x) that you do not want to be assigned dynamically by the DHCP server.
3. **(Optional)** Specify the **MAC Address** (format xx:xx:xx:xx:xx:xx) of the device to which you want to permanently assign the IP address you just specified.
4. **(Optional)** Repeat the prior two steps to reserve another address.

STEP 7 | Commit your changes.

Click **OK** and **Commit**.

Configure an Interface as a DHCP Client

Before configuring a firewall interface as a DHCP client, make sure you have configured a Layer 3 interface (Ethernet, Ethernet subinterface, VLAN, VLAN subinterface, aggregate, or aggregate subinterface) and the interface is assigned to a virtual router and a zone. Configure an interface as a DHCP client if you need to use DHCP to request an IPv4 address for the interface.



You can also [Configure the Management Interface as a DHCP Client](#).

STEP 1 | Configure an interface as a DHCP client.

1. Select **Network > Interfaces**.
2. On the **Ethernet** tab or the **VLAN** tab, **Add** a Layer 3 interface or select a configured Layer 3 interface that you want to be a DHCP client.
3. Select the **IPv4** tab and, for **Type**, select **DHCP Client**.
4. Select **Enable**.
5. (**Optional**) Enable the option to **Automatically create default route pointing to default gateway provided by server** enabled by default). Enabling this option causes the firewall to create a static route to the default gateway, which is useful when clients try to access many destinations that do not need to have routes maintained in a route table on the firewall.
6. (**Optional**) Enable the option to **Send Hostname** to assign a hostname to the DHCP client interface and send that hostname ([Option 12](#)) to a DHCP server, which can then register the hostname with the DNS server. The DNS server can then automatically manage hostname-to-dynamic IP address resolutions. External hosts can identify the interface by its hostname. The default value indicates **system-hostname**, which is the firewall hostname that you set in **Device > Setup > Management > General Settings**. Alternatively, enter a hostname for the interface, which can be a maximum of 64 characters, including uppercase and lowercase letters, numbers, period (.), hyphen (-), and underscore (_).

The screenshot shows the 'Ethernet Interface' configuration dialog. The 'IPv4' tab is selected. Under the 'Type' section, 'DHCP Client' is selected. The 'Enable' checkbox is checked. The 'Automatically create default route pointing to default gateway provided by server' checkbox is checked. The 'Send Hostname' checkbox is checked, and the value 'system-hostname' is entered in the field. The 'Default Route Metric' field is set to 10. At the bottom right, there are 'OK' and 'Cancel' buttons.

7. (**Optional**) Enter a **Default Route Metric** (priority level) for the route between the firewall and the DHCP server (range is 1 to 65,535; default is 10). A route with a lower number

has higher priority during route selection. For example, a route with a metric of 10 is used before a route with a metric of 100.



The **Default Route Metric** for the route between the firewall and the DHCP server is 10 by default. If the static default route 0.0.0.0/0 uses the DHCP interface as its egress interface, that route's **default Metric** is also 10. Therefore, there are two routes with a metric of 10 and the firewall can randomly choose one of the routes one time and the other route another time.



Suppose you enable the option to **Automatically create default route pointing to default gateway provided by server**, select a virtual router, add a static route for a Layer 3 interface, change the **Metric** (which defaults to 10) to a value greater than 10 (for this example, 100) and Commit your changes. In the route table, the route's metric will not indicate 100. Instead, it will indicate the default value of 10, as expected, because 10 takes precedence over the configured value of 100. However, if you change the static route's **Metric** to a value less than 10 (such as 6), the route in the route table is updated to indicate the configured metric of 6.

8. **(Optional)** Enable the option to **Show DHCP Client Runtime Info** to see all of the settings the client inherited from its DHCP server.

STEP 2 | Commit your changes.

Click **OK** and **Commit**.

The Ethernet interface should now indicate **Dynamic-DHCP Client** as its **IP Address** on the **Ethernet** tab.

STEP 3 | **(Optional)** See which interfaces on the firewall are configured as DHCP clients.

1. Select **Network > Interfaces > Ethernet** and check the **IP Address** to see which interfaces indicate **DHCP Client**.
2. Select **Network > Interfaces > VLAN** and check the **IP Address** to see which interfaces indicate **DHCP Client**.

Configure the Management Interface as a DHCP Client

The management interface on the firewall supports DHCP client for IPv4, which allows the management interface to receive its IPv4 address from a DHCP server. The management interface also supports DHCP Option 12 and Option 61, which allow the firewall to send its hostname and client identifier, respectively, to DHCP servers.

By default, VM-Series firewalls deployed in AWS and Azure™ use the management interface as a DHCP client to obtain its IP address, rather than a static IP address, because cloud deployments require the automation this feature provides. DHCP on the management interface is turned off by default for the VM-Series firewall except for the VM-Series firewall in AWS and Azure. The management interfaces on WildFire and Panorama models do not support this DHCP functionality.



- *For hardware-based firewall models (not VM-Series), configure the management interface with a static IP address when possible.*
- *If the firewall acquires a management interface address through DHCP, assign a MAC address reservation on the DHCP server that serves that firewall. The reservation ensures that the firewall retains its management IP address after a restart. If the DHCP server is a Palo Alto Networks® firewall, see Step 6 of [Configure an Interface as a DHCP Server](#) for reserving an address.*

If you configure the management interface as a DHCP client, the following restrictions apply:

- You cannot use the management interface in an HA configuration for control link (HA1 or HA1 backup), data link (HA2 or HA2 backup), or packet forwarding (HA3) communication.
- You cannot select **MGT** as the Source Interface when you customize service routes (**Device > Setup > Services > Service Route Configuration > Customize**). However, you can select **Use default** to route the packets via the management interface.
- You cannot use the dynamic IP address of the management interface to connect to a Hardware Security Module (HSM). The IP address on the HSM client firewall must be a static IP address because HSM authenticates the firewall using the IP address, and operations on HSM would stop working if the IP address were to change during runtime.

A prerequisite for this task is that the management interface must be able to reach a DHCP server.

STEP 1 | Configure the Management interface as a DHCP client so that it can receive its IP address (IPv4), netmask (IPv4), and default gateway from a DHCP server.

Optionally, you can also send the hostname and client identifier of the management interface to the DHCP server if the orchestration system you use accepts this information.

1. Select **Device > Setup > Management** and edit Management Interface Settings.
2. For **IP Type**, select **DHCP Client**.
3. (**Optional**) Select one or both options for the firewall to send to the DHCP server in DHCP Discover or Request messages:
 - **Send Hostname**—Sends the **Hostname** (as defined in **Device > Setup > Management**) as part of DHCP Option 12.
 - **Send Client ID**—Sends the client identifier as part of DHCP Option 61. A client identifier uniquely identifies a DHCP client, and the DHCP Server uses it to index its configuration parameter database.
4. Click **OK**.

STEP 2 | (**Optional**) Configure the firewall to accept the host name and domain from the DHCP server.

1. Select **Device > Setup > Management** and edit General Settings.
2. Select one or both options:
 - **Accept DHCP server provided Hostname**—Allows the firewall to accept the hostname from the DHCP server (if valid). When enabled, the hostname from the DHCP server overwrites any existing **Hostname** specified in **Device > Setup > Management**. Don't select this option if you want to manually configure a hostname.
 - **Accept DHCP server provided Domain**—Allows the firewall to accept the domain from the DHCP Server. The domain (DNS suffix) from the DHCP Server overwrites any existing **Domain** specified in **Device > Setup > Management**. Don't select this option if you want to manually configure a domain.
3. Click **OK**.

STEP 3 | Commit your changes.

Click **Commit**.

STEP 4 | View DHCP client information.

1. Select **Device > Setup > Management** and Management Interface Settings.
2. Click **Show DHCP Client Runtime Info**.

STEP 5 | (**Optional**) Renew the **DHCP lease** with the DHCP server, regardless of the lease term.

This option is convenient if you are testing or troubleshooting network issues.

1. Select **Device > Setup > Management** and edit Management Interface Settings.
2. Click **Show DHCP Client Runtime Info**.
3. Click **Renew**.

STEP 6 | (Optional) Release the following DHCP options that came from the DHCP server:

- IP Address
- Netmask
- Default Gateway
- DNS Server (primary and secondary)
- NTP Server (primary and secondary)
- Domain (DNS Suffix)

 A release frees the IP address, which drops your network connection and renders the firewall unmanageable if no other interface is configured for management access.

Use the CLI operational command **request dhcp client management-interface release**.

Configure an Interface as a DHCP Relay Agent

To enable a firewall interface to transmit [DHCP messages between clients and servers](#), you must configure the firewall as a DHCP relay agent. The interface can forward messages to a maximum of eight external IPv4 DHCP servers and eight external IPv6 DHCP servers. A client DHCPDISCOVER message is sent to all configured servers, and the DHCPOFFER message of the first server that responds is relayed back to the requesting client.

Capacities are as follows:

- You can configure a combined total of 500 DHCP servers (IPv4) and DHCP relay agents (IPv4 and IPv6) on all firewall models except for PA-5200 Series and PA-7000 Series firewalls
- On PA-5220 firewalls, you can configure a maximum of 500 DHCP servers and a maximum of 2,048 DHCP relay agents minus the number of DHCP servers configured. For example, if you configure 500 DHCP servers, you can configure 1,548 DHCP relay agents.
- On PA-5250, PA-5260, and PA-7000 Series firewalls, you can configure a maximum of 500 DHCP servers, and a maximum of 4,096 DHCP relay agents minus the number of DHCP servers configured. For example, if you configure 500 DHCP servers, you can configure 3,596 DHCP relay agents.

Before configuring a DHCP relay agent, make sure you have configured a Layer 3 Ethernet or Layer 3 VLAN interface, and the interface is assigned to a virtual router and a zone.

STEP 1 | Select DHCP Relay.

Select Network > DHCP > DHCP Relay.

STEP 2 | Specify the IP address of each DHCP server with which the DHCP relay agent will communicate.

1. In the **Interface** field, select the interface you want to be the DHCP relay agent.
2. Select either **IPv4** or **IPv6**, indicating the type of DHCP server address you will specify.
3. If you checked **IPv4**, in the **DHCP Server IP Address** field, **Add** the address of the DHCP server to and from which you will relay DHCP messages.
4. If you checked **IPv6**, in the **DHCP Server IPv6 Address** field, **Add** the address of the DHCP server to and from which you will relay DHCP messages. If you specify a *multicast* address, also specify an outgoing **Interface**.
5. (**Optional**) Repeat the prior three steps to enter a maximum of eight DHCP server addresses per IP address family.

STEP 3 | Commit the configuration.

Click **OK** and **Commit**.

Monitor and Troubleshoot DHCP

You can view the status of dynamic address leases that your DHCP server has assigned or that your DHCP client has been assigned by issuing commands from the CLI. You can also clear leases before they time out and are released automatically.

- [View DHCP Server Information](#)
- [Clear DHCP Leases](#)
- [View DHCP Client Information](#)
- [Gather Debug Output about DHCP](#)

View DHCP Server Information

Perform this task to view DHCP pool statistics, IP addresses the DHCP server has assigned, the corresponding MAC address, state and duration of the lease, and time the lease began. If the address was configured as a **Reserved Address**, the **state** column indicates reserved and there is no **duration** or **lease_time**. If the lease was configured as **Unlimited**, the **duration** column displays a value of 0.

- View DHCP pool statistics, IP address the DHCP server assigned, MAC address, state and duration of lease, and lease start time.

```
admin@PA-220> show dhcp server lease interface all
```

```
interface: "ethernet1/2"
Allocated IPs: 1, Total number of IPs in pool: 5. 20.0000% used
ip           mac           state      duration
lease_time
192.168.3.11   f0:2f:af:42:70:cf  committed  0          Wed Jul
2 08:10:56 2014
admin@PA-220>
```

- View the options that a DHCP server has assigned to clients.

```
admin@PA-220> show dhcp server settings all
```

Interface	GW	DNS1	DNS2	DNS-Suffix	Inherit
source					

ethernet1/2	192.168.3.1	10.43.2.10	10.44.2.10		
	ethernet1/3				

```
admin@PA-220>
```

Clear DHCP Leases

You have several options for clearing DHCP leases.

DHCP

- Release expired **DHCP Leases** of an interface (server), such as ethernet1/2, before the hold timer releases them automatically. Those addresses will be available in the IP pool again.

```
admin@PA-220> clear dhcp lease interface ethernet1/2 expired-only
```

- Release the lease of a particular IP address, for example, 192.168.3.1.

```
admin@PA-220> clear dhcp lease interface ethernet1/2 ip 192.168.3.1
```

- Release the lease of a particular MAC address, for example, f0:2c:ae:29:71:34.

```
admin@PA-220> clear dhcp lease interface ethernet1/2 mac  
f0:2c:ae:29:71:34
```

View DHCP Client Information

To view the status of IP address leases sent to the firewall when it is acting as a DHCP client, use either of these CLI commands.

- admin@PA-220>**show dhcp client state <interface_name>**
- admin@PA-220> **show dhcp client state all**

Interface Leased-until	State	IP	Gateway
ethernet1/1 70315	Bound	10.43.14.80	10.43.14.1

Gather Debug Output about DHCP

To gather debug output about DHCP, use one of the following commands:

- admin@PA-220> **debug dhcpd**
- admin@PA-220> **debug management-server dhcpd**

DNS

Domain Name System (DNS) is a protocol that translates (resolves) a user-friendly domain name, such as www.paloaltonetworks.com, to an IP address so that users can access computers, websites, services, or other resources on the internet or private networks.

- [DNS Overview](#)
- [DNS Proxy Object](#)
- [DNS Server Profile](#)
- [Multi-Tenant DNS Deployments](#)
- [Configure a DNS Proxy Object](#)
- [Configure a DNS Server Profile](#)
- [Use Case 1: Firewall Requires DNS Resolution](#)
- [Use Case 2: ISP Tenant Uses DNS Proxy to Handle DNS Resolution for Security Policies, Reporting, and Services within its Virtual System](#)
- [Use Case 3: Firewall Acts as DNS Proxy Between Client and Server](#)
- [DNS Proxy Rule and FQDN Matching](#)

DNS Overview

DNS performs a crucial role in enabling user access to network resources so that users need not remember IP addresses and individual computers need not store a huge volume of domain names mapped to IP addresses. DNS employs a client/server model; a DNS server resolves a query for a DNS client by looking up the domain in its cache and if necessary sending queries to other servers until it can respond to the client with the corresponding IP address.

The DNS structure of domain names is hierarchical; the top-level domain (TLD) in a domain name can be a generic TLD (gTLD): com, edu, gov, int, mil, net, or org (gov and mil are for the United States only) or a country code (ccTLD), such as au (Australia) or us (United States). ccTLDs are generally reserved for countries and dependent territories.

A fully qualified domain name (FQDN) includes at a minimum a host name, a second-level domain, and a TLD to completely specify the location of the host in the DNS structure. For example, www.paloaltonetworks.com is an FQDN.

Wherever a Palo Alto Networks[®] firewall uses an FQDN in the user interface or CLI, the firewall must resolve that FQDN using DNS. Depending on where the FQDN query originates, the firewall determines which DNS settings to use to resolve the query.

A DNS record of an FQDN includes a time-to-live (TTL) value, and by default the firewall refreshes each FQDN in its cache based on that individual TTL provided the DNS server, as long as the TTL is greater than or equal to the [Minimum FQDN Refresh Time](#) you configure on the firewall, or the default setting of 30 seconds if you don't configure a minimum. Refreshing an FQDN based on its TTL value is especially helpful for securing access to cloud platform services, which often require frequent FQDN refreshes to ensure highly available services. For example, cloud environments that support autoscaling depend on FQDN resolutions for dynamically scaling services up and down, and fast resolutions of FQDNs are critical in such time-sensitive environments.

By configuring a minimum FQDN refresh time, you limit how small a TTL value the firewall honors. If your IP addresses don't change very often you may want to set a higher Minimum FQDN Refresh Time so that the firewall doesn't refresh entries unnecessarily. The firewall uses the higher of the DNS TTL time and the configured Minimum FQDN Refresh Time.

For example, two FQDNs have the following TTL values. The Minimum FQDN Refresh Time overrides smaller (faster) TTL values.

	TTL	If Minimum FQDN Refresh = 26	Actual Refresh Time
FQDN A	20		26
FQDN B	30		30

The FQDN refresh timer starts when the firewall receives a DNS response from the DNS server or DNS proxy object that is resolving the FQDN.

Additionally, you can set a [stale timeout](#) to configure how long the firewall continues to use stale (expired) FQDN resolutions in the event of an unreachable DNS Server. At the end of the stale

timeout period, if the DNS server is still unreachable, the stale FQDN entries become unresolved (the firewall removes stale FQDN entries).

The following firewall tasks are related to DNS:

- Configure your firewall with at least one DNS server so it can resolve hostnames. Configure primary and secondary DNS servers or a DNS Proxy object that specifies such servers, as shown in [Use Case 1: Firewall Requires DNS Resolution](#).
- Customize how the firewall handles DNS resolution initiated by Security policy rules, reporting, and management services (such as email, Kerberos, SNMP, syslog, and more) for each virtual system, as shown in [Use Case 2: ISP Tenant Uses DNS Proxy to Handle DNS Resolution for Security Policies, Reporting, and Services within its Virtual System](#).
- Configure the firewall to act as a DNS server for a client, as shown in [Use Case 3: Firewall Acts as DNS Proxy Between Client and Server](#).
- Configure an Anti-Spyware profile to [Use DNS Queries to Identify Infected Hosts on the Network](#).
- [Enable Evasion Signatures](#) and then enable evasion signatures for threat prevention.
- [Configure an Interface as a DHCP Server](#). This enables the firewall to act as a DHCP Server and sends DNS information to its DHCP clients so the provisioned DHCP clients can reach their respective DNS servers.

DNS Proxy Object

When configured as a DNS proxy, the firewall is an intermediary between DNS clients and servers; it acts as a DNS server itself by resolving queries from its DNS proxy cache. If it doesn't find the domain name in its DNS proxy cache, the firewall searches for a match to the domain name among the entries in the specific DNS proxy object (on the interface on which the DNS query arrived). The firewall forwards the query to the appropriate DNS server based on the match results. If no match is found, the firewall uses default DNS servers.

A DNS proxy object is where you configure the settings that determine how the firewall functions as a DNS proxy. You can assign a DNS proxy object to a single virtual system or it can be shared among all virtual systems.

- If the DNS proxy object is for a virtual system, you can specify a [DNS Server Profile](#), which specifies the primary and secondary DNS server addresses, along with other information. The DNS server profile simplifies configuration.
- If the DNS proxy object is shared, you must specify at least the primary address of a DNS server.



When configuring multiple tenants (ISP subscribers) with DNS services, each tenant should have its own DNS proxy defined, which keeps the tenant's DNS service separate from other tenants' services.

In the proxy object, you specify the interfaces for which the firewall is acting as DNS proxy. The DNS proxy for the interface does not use the service route; responses to the DNS requests are always sent to the interface assigned to the virtual router where the DNS request arrived.

When you [Configure a DNS Proxy Object](#), you can supply the DNS proxy with static FQDN-to-address mappings. You can also create DNS proxy rules that control to which DNS server the domain name queries (that match the proxy rules) are directed. You can configure a maximum of 256 DNS proxy objects on a firewall. You must enable **Cache** and **Cache EDNS Responses** (under **Network > DNS Proxy > Advanced**) if this DNS proxy object is assigned to **Device > Setup > Services > DNS** or **Device > Virtual Systems > vsys > General > DNS Proxy**. Furthermore, if this DNS proxy object has **DNS proxy rules** configured, those rules also need to have cache enabled (**Turn on caching of domains resolved by this mapping**).

When the firewall receives an FQDN query (and the domain name is not in the DNS proxy cache), the firewall compares the domain name from the FQDN query to the domain names in DNS Proxy rules of the DNS Proxy object. If you specify multiple domain names in a single DNS Proxy rule, a query that matches any one of the domain names in the rule means the query matches the rule. [DNS Proxy Rule and FQDN Matching](#) describes how the firewall determines whether an FQDN matches a domain name in a DNS proxy rule. A DNS query that matches a rule is sent to the primary DNS server configured for the proxy object to be resolved.

DNS Server Profile

To simplify configuration for a virtual system, a DNS server profile allows you to specify the virtual system that is being configured, an inheritance source or the primary and secondary IP addresses for DNS servers, and a source interface and source address (service route) that will be used in packets sent to the DNS server. The source interface determines the virtual router, which has a route table. The destination IP address is looked up in the route table of the virtual router where the source interface is assigned. It's possible that the result of the destination IP egress interface differs from the source interface. The packet would egress out of the destination IP egress interface determined by the route table lookup, but the source IP address would be the address configured. The source address is used as the destination address in the reply from the DNS server.

The virtual system report and virtual system server profile send their queries to the DNS server specified for the virtual system, if there is one. (The DNS server used is defined in **Device > Virtual Systems > General > DNS Proxy**.) If there is no DNS server specified for the virtual system, the DNS server specified for the firewall is queried.

You [Configure a DNS Server Profile](#) for a virtual system only; it is not for a global Shared location.

Multi-Tenant DNS Deployments

The firewall determines how to handle DNS requests based on where the request originated. An environment where an ISP has multiple tenants on a firewall is known as multi-tenancy. There are three use cases for multi-tenant DNS deployments:

- **Global Management DNS Resolution**—The firewall needs DNS resolution for its own purposes, for example, the request comes from the management plane to resolve an FQDN for a management event such as a software update service. The firewall uses the service route to get to a DNS server because DNS request isn't coming in on a specific virtual router.
- **Policy and Report FQDN Resolution for a Virtual System**—For DNS queries from a security policy, a report, or a service, you can specify a set of DNS servers specific to the virtual system (tenant) or you can default to the global DNS servers. If your use case requires a different set of DNS servers per virtual system, you must configure a [DNS Proxy Object](#). The resolution is specific to the virtual system to which the DNS proxy is assigned. If you don't have specific DNS servers applicable to this virtual system, the firewall uses the global DNS settings.
- **Dataplane DNS Resolution for a Virtual System**—This method is also known as a Network Request for DNS Resolution. The tenant's virtual system can be configured so that specified domain names are resolved on the tenant's DNS server in its network. This method supports *split DNS*, meaning that the tenant can also use its own ISP DNS servers for the remaining DNS queries not resolved on its own server. [DNS Proxy Object](#) rules control the split DNS; the tenant's domain redirects DNS requests to its DNS servers, which are configured in a DNS server profile. The DNS server profile has primary and secondary DNS servers designated, and also DNS service routes for IPv4 and IPv6, which override the default DNS settings.

The following table summarizes the DNS resolution types. The binding location determines which DNS proxy object is used for the resolution. For illustration purposes, the use cases show how a service provider might configure DNS settings to provide DNS services for resolving DNS queries required on the firewall and for tenant (subscriber) virtual systems.

Resolution Type	Location: Shared	Location: Specific Vsys
Firewall DNS resolution—performed by management plane	Binding: Global Illustrated in Use Case 1	N/A
Security profile, reporting, and server profile resolution—performed by management plane	Binding: Global Same behavior as Use Case 1	Binding: Specific vsys Illustrated in Use Case 2
DNS proxy resolution for DNS client hosts connected to interface on firewall, going through the firewall to a DNS Server—performed by dataplane	Binding: Interface Service Route: Interface and IP address on which the DNS Request was received. Illustrated in Use Case 3	

- Use Case 1: Firewall Requires DNS Resolution
- Use Case 2: ISP Tenant Uses DNS Proxy to Handle DNS Resolution for Security Policies, Reporting, and Services within its Virtual System
- Use Case 3: Firewall Acts as DNS Proxy Between Client and Server

Configure a DNS Proxy Object

If your firewall is to act as a DNS proxy, perform this task to configure a [DNS Proxy Object](#). The proxy object can either be shared among all virtual systems or applied to a specific virtual system.



When the firewall is enabled to act as a DNS proxy, evasion signatures that detected crafted HTTP or TLS requests can alert to instances where a client connects to a domain other than the domains specified in the original DNS query. As a best practice, [Enable Evasion Signatures](#) after configuring DNS proxy to trigger an alert if crafted requests are detected.

STEP 1 | Configure the basic settings for a DNS Proxy object.

1. Select **Network > DNS Proxy** and **Add** a new object.
2. Verify that **Enable** is selected.
3. Enter a **Name** for the object.
4. For **Location**, select the virtual system to which the object applies. If you select **Shared**, you must specify at least a **Primary** DNS server address, and optionally a **Secondary** address.
5. If you selected a virtual system, for **Server Profile**, select a DNS Server profile or else click **DNS Server Profile** to configure a new profile. See [Configure a DNS Server Profile](#).
6. For Inheritance Source, select a source from which to inherit default DNS server settings. The default is **None**.
7. For **Interface**, click **Add** and specify the interfaces to which the DNS Proxy object applies.
 - If you use the DNS Proxy object for performing DNS lookups, an interface is required. The firewall will listen for DNS requests on this interface, and then proxy them.
 - If you use the DNS Proxy object for a service route, the interface is optional.

STEP 2 | (Optional) Specify DNS Proxy rules.

1. On the **DNS Proxy Rules** tab, **Add** a **Name** for the rule.
2. **Turn on caching of domains resolved by this mapping** if you want the firewall to cache the resolved domains.
3. For **Domain Name**, **Add** one or more domains, one entry per row, to which the firewall compares FQDN queries. If a query matches one of the domains in the rule, the query is

sent to one of the following servers to be resolved (depending on what you configured in the prior step):

- The **Primary** or **Secondary** DNS Server directly specified for this proxy object.
- The **Primary** or **Secondary** DNS Server specified in the DNS Server profile for this proxy object.

[DNS Proxy Rule and FQDN Matching](#) describes how the firewall matches domain names in an FQDN to a DNS proxy rule. If no match is found, default DNS servers resolve the query.

4. Do one of the following, depending on what you set the **Location** to:
 - If you chose a virtual system, select a **DNS Server profile**.
 - If you chose **Shared**, enter a **Primary** and optionally a **Secondary** address.
5. Click **OK**.

STEP 3 | [\(Optional\)](#) Supply the DNS Proxy with static FQDN-to-address entries. Static DNS entries allow the firewall to resolve the FQDN to an IP address without sending a query to the DNS server.

1. On the **Static Entries** tab, **Add a Name**.
2. Enter the Fully Qualified Domain Name (**FQDN**).
3. For **Address**, Add the IP address to which the FQDN should be mapped.

You can provide additional IP addresses for an entry. The firewall will provide all of the IP addresses in its DNS response and the client chooses which address to use.

4. Click **OK**.

STEP 4 | Enable caching and configure other advanced settings for the DNS Proxy.

1. On the **Advanced** tab, select **TCP Queries** to enable DNS queries using TCP.
 - **Max Pending Requests**—Enter the maximum number of concurrent, pending TCP DNS requests that the firewall will support (range is 64-256; default is 64).
2. For **UDP Queries Retries**, enter:
 - **Interval (sec)**—The length of time (in seconds) after which another request is sent if no response has been received (range is 1 to 30; default is 2).
 - **Attempts**—The maximum number of UDP query attempts (excluding the first attempt) after which the next DNS server is queried (range is 1 to 30; default is 5.)
3. Select **Cache** to enable the firewall to cache FQDN-to-address mappings that it learns. You must enable **Cache** (enabled by default) if this DNS proxy object is used for queries that the firewall generates (that is, under **Device > Setup > Services > DNS**, or under **Device > Virtual Systems** and you select a virtual system and **General > DNS Proxy**).
 - Select **Enable TTL** to limit the length of time the firewall caches DNS resolution entries for the proxy object. Disabled by default.
 - Enter **Time to Live (sec)**, the number of seconds after which all cached entries for the proxy object are removed. After the entries are removed, new DNS requests

must be resolved and cached again. Range is 60-86,400. There is no default TTL; entries remain until the firewall runs out of cache memory.

- **Cache EDNS Responses**—You must enable this setting if this DNS proxy object is used for queries that the firewall generates (that is, under **Device > Setup > Services > DNS**, or under **Device > Virtual Systems** and you select a virtual system and **General > DNS Proxy**).

STEP 5 | Commit your changes.

Click **OK** and **Commit**.

Configure a DNS Server Profile

Configure a [DNS Server Profile](#), which simplifies configuration of a virtual system. The **Primary DNS** or **Secondary DNS** address is used to create the DNS request that the virtual system sends to the DNS server.

STEP 1 | Name the DNS server profile, select the virtual system to which it applies, and specify the primary and secondary DNS server addresses.

1. Select **Device > Server Profiles > DNS** and Add a **Name** for the DNS server profile.
2. For **Location**, select the virtual system to which the profile applies.
3. For **Inheritance Source**, select **None** if the DNS server addresses are not inherited. Otherwise, specify the DNS server from which the profile should inherit settings. If you choose a DNS server, click **Check inheritance source status** to see that information.
4. Specify the IP address of the **Primary DNS** server, or leave as **inherited** if you chose an **Inheritance Source**.



*Keep in mind that if you specify an FQDN instead of an IP address, the DNS for that FQDN is resolved in **Device > Virtual Systems > DNS Proxy**.*

5. Specify the IP address of the **Secondary DNS** server, or leave as **inherited** if you chose an **Inheritance Source**.

STEP 2 | Configure the service route that the firewall automatically uses, based on whether the target DNS Server has an IP address family type of IPv4 or IPv6.

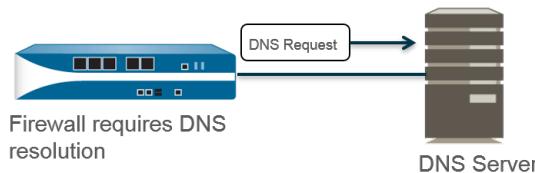
1. Click **Service Route IPv4** to enable the subsequent interface and IPv4 address to be used as the service route, if the target DNS address is an IPv4 address.
2. Specify the **Source Interface** to select the DNS server's source IP address that the service route will use. The firewall determines which virtual router is assigned that interface, and then does a route lookup in the virtual router routing table to reach the destination network (based on the **Primary DNS** address).
3. Specify the **IPv4 Source Address** from which packets going to the DNS server are sourced.
4. Click **Service Route IPv6** to enable the subsequent interface and IPv6 address to be used as the service route, if the target DNS address is an IPv6 address.
5. Specify the **Source Interface** to select the DNS server's source IP address that the service route will use. The firewall determines which virtual router is assigned that interface, and then does a route lookup in the virtual router routing table to reach the destination network (based on the **Primary DNS** address).
6. Specify the **IPv6 Source Address** from which packets going to the DNS server are sourced.
7. Click **OK**.

STEP 3 | Commit the configuration.

Click **OK** and **Commit**.

Use Case 1: Firewall Requires DNS Resolution

In this use case, the firewall is the client requesting DNS resolutions of FQDNs for Security policy rules, reporting, management services (such as email, Kerberos, SNMP, syslog, and more), and management events such as software update services, dynamic software updates, and WildFire. In dynamic environments, FQDNs change more frequently; accurate DNS resolutions allow the firewall to enforce accurate policing, provide reporting and management services, and handle management events. The shared, global DNS services perform the DNS resolution for the management plane functions.



STEP 1 | Configure the primary and secondary DNS servers you want the firewall to use for DNS resolutions.

 You must manually configure at least one DNS server on the firewall or it won't be able to resolve hostnames; the firewall cannot use DNS server settings from another source, such as an ISP.

1. Edit the Services settings (Device > Setup > Services > Global for firewalls that support multiple virtual systems; Device > Setup > Services for those that don't).
2. On the **Services** tab, for **DNS**, select **Servers** and enter the **Primary DNS Server** address and **Secondary DNS Server** address.
3. Proceed to Step 3.

STEP 2 | Alternatively, you can configure a **DNS Proxy Object** if you want to configure advanced DNS functions such as split DNS, DNS proxy overrides, DNS proxy rules, static entries, or DNS inheritance.

1. Edit the Services settings (**Device > Setup > Services > Global** for firewalls that support multiple virtual systems; **Device > Setup > Services** for those that don't).
2. On the **Services** tab, for **DNS**, select **DNS Proxy Object**.
3. From the **DNS Proxy** list, select the DNS proxy that you want to use to configure global DNS services, or select **DNS Proxy** to configure a new DNS proxy object as follows:
 1. **Enable** and then enter a **Name** for the DNS proxy object.
 2. On firewalls that support multiple virtual systems, for **Location**, select **Shared** for global, firewall-wide DNS proxy services.

 *Shared DNS proxy objects don't use DNS server profiles because they don't require a specific service route belonging to a tenant virtual system.*

3. Enter the **Primary** DNS server IP address. Optionally enter a **Secondary** DNS server IP address.
4. Select the **Advanced** tab. Ensure that **Cache** is enabled and **Cache EDNS Responses** is enabled (both are enabled by default).
5. Click **OK** to save the DNS Proxy object.

STEP 3 | **(Optional)** Set a **Minimum FQDN Refresh Time (sec)** to limit how frequently the firewall refreshes FQDN cache entries.

By default, the firewall refreshes each FQDN in its cache based on the individual TTL for the **FQDN in a DNS record**, as long as the TTL is greater than or equal to this minimum FQDN refresh setting (or as long as the TTL is greater than or equal to the default setting of 30 seconds if you don't configure a minimum FQDN refresh time). To set a minimum FQDN refresh time, enter a value in seconds (range is 0 to 14,400; default is 30). A setting of 0 means the firewall refreshes FQDNs based on the TTL value in the DNS records; the firewall doesn't enforce a minimum FQDN refresh time. The firewall uses the higher of the DNS TTL time and the minimum FQDN refresh time.

 *If the TTL for the FQDN in DNS is short, but your FQDN resolutions don't change as frequently as the TTL timeframe so don't need a faster refresh, you should set a Minimum FQDN Refresh Time to avoid making FQDN refresh attempts more often than necessary.*

STEP 4 | **(Optional)** Specify an **FQDN Stale Entry Timeout (min)**, which is the number of minutes that the firewall continues to use stale FQDN resolutions in the event of an unreachable DNS server (range is 0 to 10,080; default is 1,440).

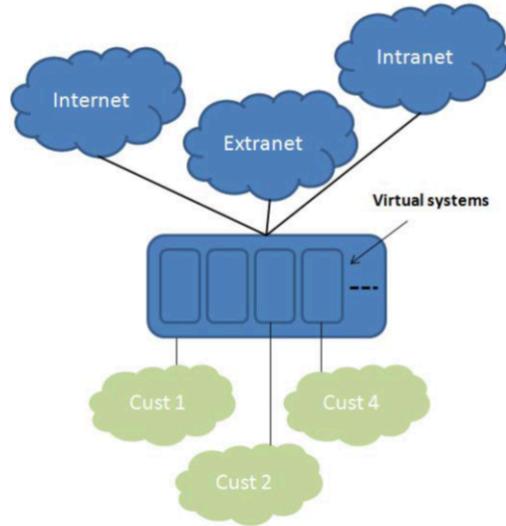
A setting of 0 means the firewall does not continue to use a stale FQDN entry.

 *Make sure the FQDN stale entry timeout is short enough not to allow incorrect traffic forwarding (which can pose a security risk), but long enough to allow traffic continuity without causing an unplanned network outage.*

STEP 5 | Click **OK** and **Commit**.

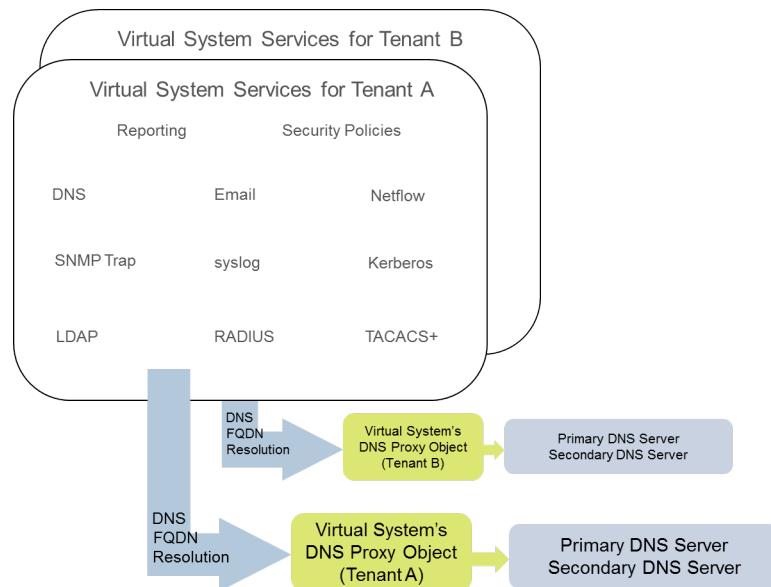
Use Case 2: ISP Tenant Uses DNS Proxy to Handle DNS Resolution for Security Policies, Reporting, and Services within its Virtual System

In this use case, multiple tenants (ISP subscribers) are defined on the firewall and each tenant is allocated a separate virtual system (vsys) and virtual router in order to segment its services and administrative domains. The following figure illustrates several virtual systems within a firewall.



Each tenant has its own server profiles for Security policy rules, reporting, and management services (such as email, Kerberos, SNMP, syslog, and more) defined in its own networks.

For the DNS resolutions initiated by these services, each virtual system is configured with its own [DNS Proxy Object](#) to allow each tenant to customize how DNS resolution is handled within its virtual system. Any service with a **Location** will use the DNS Proxy object configured for the virtual system to determine the primary (or secondary) DNS server to resolve FQDNs, as illustrated in the following figure.



STEP 1 | For each virtual system, specify the DNS Proxy to use.

1. Select **Device > Virtual Systems** and Add the **ID** of the virtual system (range is 1-255), and an optional **Name**, in this example, Corp1 Corporation.
2. On the **General** tab, choose a **DNS Proxy** or create a new one. In this example, Corp1 DNS Proxy is selected as the proxy for Corp1 Corporation's virtual system.
3. For **Interfaces**, click **Add**. In this example, Ethernet1/20 is dedicated to this tenant.
4. For **Virtual Routers**, click **Add**. A virtual router named Corp1 VR is assigned to the virtual system in order to separate routing functions.
5. Click **OK**.

STEP 2 | Configure a DNS Proxy and a server profile to support DNS resolution for a virtual system.

1. Select **Network > DNS Proxy** and click **Add**.
2. Click **Enable** and enter a **Name** for the DNS Proxy.
3. For **Location**, select the virtual system of the tenant, in this example, Corp1 Corporation (vsys6). (You could choose the **Shared** DNS Proxy resource instead.)
4. For **Server Profile**, choose or create a profile to customize DNS servers to use for DNS resolutions for this tenant's security policy, reporting, and server profile services.

If the profile is not already configured, in the **Server Profile** field, click **DNS Server Profile** to [Configure a DNS Server Profile](#).

The DNS server profile identifies the IP addresses of the primary and secondary DNS server to use for management DNS resolutions for this virtual system.

5. Also for this server profile, optionally configure a **Service Route IPv4** and/or a **Service Route IPv6** to instruct the firewall which **Source Interface** to use in its DNS requests. If that interface has more than one IP address, configure the **Source Address** also.
6. Select the **Advanced** tab. Ensure that **Cache** is enabled and **Cache EDNS Responses** is enabled (both are enabled by default). This is required if the DNS proxy object is used under **Device > Virtual Systems > vsys > General > DNS Proxy**.
7. Click **OK**.
8. Click **OK** and **Commit**.



*Optional advanced features such as split DNS can be configured using **DNS Proxy Rules**. A separate DNS server profile can be used to redirect DNS resolutions matching the **Domain Name** in a **DNS Proxy Rule** to another set of DNS servers, if required. Use Case 3 illustrates split DNS.*

If you use two separate DNS server profiles in the same DNS Proxy object, one for the DNS Proxy and one for the DNS proxy rule, the following behaviors occur:

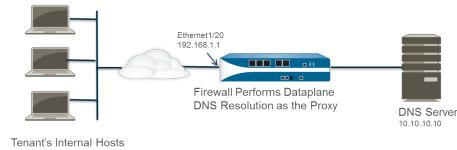
- If a service route is defined in the DNS server profile used by the DNS Proxy, it takes precedence and is used.
- If a service route is defined in the DNS server profile used in the DNS proxy rules, it is not used. If the service route differs from the one defined in the DNS server profile used by the DNS Proxy, the following warning message is displayed during the **Commit** process:

Warning: The DNS service route defined in the DNS proxy object is different from the DNS proxy rule's service route. Using the DNS proxy object's service route.

- If no service route is defined in any DNS server profile, the global service route is used if needed.

Use Case 3: Firewall Acts as DNS Proxy Between Client and Server

In this use case, the firewall is located between a DNS client and a DNS server. A DNS Proxy on the firewall is configured to act as the DNS server for the hosts that reside on the tenant's network connected to the firewall interface. In such a scenario, the firewall performs DNS resolution on its dataplane.



This scenario happens to use *split DNS*, a configuration where DNS Proxy rules are configured to redirect DNS requests to a set of DNS servers based on a domain name match. If there is no match, the server profile determines the DNS servers to which to send the request, hence the two, split DNS resolution methods.



For dataplane DNS resolutions, the source IP address from the DNS proxy in PAN-OS to the outside DNS server would be the address of the proxy (the destination IP of the original request). Any service routes defined in the DNS Server Profile are not used. For example, if the request is from host 172.16.1.1 to the DNS proxy at 192.168.1.1, then the request to the DNS server (at 10.10.10.10) would use a source of 192.168.1.1 and a destination of 10.10.10.10.

STEP 1 | Select **Network > DNS Proxy** and click **Add**.

STEP 2 | Click **Enable** and enter a **Name** for the DNS Proxy.

STEP 3 | For **Location**, select the virtual system of the tenant, in this example, Corp1 Corporation (**vsys6**).

STEP 4 | For **Interface**, select the interface that will receive the DNS requests from the tenant's hosts, in this example, **Ethernet1/20**.

STEP 5 | Choose or create a **Server Profile** to customize DNS servers to resolve DNS requests for this tenant.

STEP 6 | On the **DNS Proxy Rules** tab, Add a **Name** for the rule.

STEP 7 | (Optional) Select **Turn on caching of domains resolved by this mapping**.

STEP 8 | Add one or more **Domain Name(s)**, one entry per row. [DNS Proxy Rule and FQDN Matching](#) describes how the firewall matches FQDNs to domain names in a DNS proxy rule.

STEP 9 | For **DNS Server profile**, select a profile. The firewall compares the domain name in the DNS request to the domain name(s) defined in the **DNS Proxy Rules**. If there is a match, the **DNS Server profile** defined in the rule is used to determine the DNS server.

STEP 10 | In this example, if the domain in the request matches myweb.corp1.com, the DNS server defined in the myweb DNS Server Profile is used. If there is no match, the DNS server defined in the **Server Profile** (Corp1 DNS Server Profile) is used.

STEP 11 | Click **OK** twice.

DNS Proxy Rule and FQDN Matching

When you configure the firewall with a [DNS Proxy Object](#) that uses DNS proxy rules, the firewall compares an FQDN from a DNS query to the domain name of a DNS proxy rule. The firewall comparison works as follows:

FQDN Comparison to DNS Proxy Rule	For Example
The firewall first tokenizes the FQDNs and the domain names in the DNS proxy rules. In a domain name, a string delimited by a period (.) is a token.	*. boat . fish . com consists of four tokens: [*] [boat][fish][com]
The matching process is an exact token match between the FQDN and the domain name in the rule; partial strings are not matched.	Rule: fishing FQDN: fish – Not a Match
An exception to the exact match requirement is the use of the wildcard –an asterisk (*). The * matches one or more tokens. This means a rule consisting of only a wildcard (*) matches any FQDN with one or more tokens.	Rule: *. boat . com FQDN: www.boat.com – Match FQDN: www.blue.boat.com – Match FQDN: boat.com – Not a Match
You can use an * in any position: preceding tokens, between tokens, or trailing tokens (but not with other characters within a single token).	Rule: * FQDN: boat – Match FQDN: boat.com – Match FQDN: www.boat.com – Match
	Rule: www.boat.* FQDN: www.boat.com – Match FQDN: www.boat.fish.com – Match
	Rule: www.boat*.com – Invalid
Multiple wildcards (*) can appear in any position of the domain name: preceding tokens, between tokens, or trailing	Rule: a.*.d.*.com FQDN: a.b.d.e.com – Match FQDN: a.b.c.d.e.f.com – Match

FQDN Comparison to DNS Proxy Rule	For Example
tokens. Each non-consecutive * matches one or more tokens.	FQDN: a.d.d.e.f.com – Match (First * matches d ; second * matches e and f) FQDN: a.d.e.f.com – Not a Match (First * matches d ; subsequent d in the rule is not matched)
When wildcards are used in consecutive tokens, the first * matches one or more tokens; the second * matches one token. This means a rule consisting of only *.* matches any FQDN with two or more tokens.	Consecutive wildcards preceding tokens: Rule: *.*.boat.com FQDN: www.blue.boat.com – Match FQDN: www.blue.sail.boat.com – Match
	Consecutive wildcards between tokens: Rule: www.*.*.boat.com FQDN: www.blue.sail.boat.com – Match FQDN: www.big.blue.sail.boat.com – Match
	Consecutive wildcards trailing tokens: Rule: www.boat.*.* FQDN: www.boat.fish.com – Match FQDN: www.boat.fish.ocean.com – Match
	Consecutive wildcards only: Rule: *.* FQDN: boat – Not a Match FQDN: boat.com – Match FQDN: www.boat.com – Match
Consecutive and non-consecutive wildcards can appear in the same rule.	Rule: a.*.d.*.*.com FQDN: a.b.c.d.e.f.com – Match (First * matches b and c ; second * matches e ; third * matches f) FQDN: a.b.c.d.e.com – Not a Match (First * matches b and c ; second * matches e ; third * not matched)
The Implicit-tail-match behavior provides an additional shorthand:	Rule: www.boat.fish

FQDN Comparison to DNS Proxy Rule	For Example
As long as the last token of the rule is not an *, a comparison will match if all tokens in the rule match the FQDN, even when the FQDN has additional trailing tokens that the rule doesn't have.	<p>FQDN: www.boat.fish.com – Match</p> <p>FQDN: www.boat.fish.ocean.com – Match</p> <p>FQDN: www.boat.fish – Match</p>
This rule ends with *, so the Implicit-tail-match rule doesn't apply. The * behaves as stated; it matches one or more tokens.	<p>Rule: www.boat.fish.*</p> <p>FQDN: www.boat.fish.com – Match</p> <p>FQDN: www.boat.fish.ocean.com – Match</p> <p>FQDN: www.boat.fish – Not a Match (This FQDN does not have a token to match the * in the rule.)</p>
In the case where an FQDN matches more than one rule, a tie-breaking algorithm selects the most specific (longest) rule; that is, the algorithm favors the rule with more tokens and fewer wildcards (*).	<p>Rule 1: *.fish.com – Match</p> <p>Rule 2: *.com – Match</p> <p>Rule 3: boat.fish.com – Match and Tie-Breaker</p> <p>FQDN: boat.fish.com</p> <p>FQDN matches all three rules; the firewall uses Rule 3 because it is the most specific.</p>
	<p>Rule 1: *.fish.com – Not a Match</p> <p>Rule 2: *.com – Match</p> <p>Rule 3: boat.fish.com – Not a Match</p> <p>FQDN: fish.com</p> <p>FQDN does not match Rule 1 because the * does not have a token to match.</p>
	<p>Rule 1: *.fish.com – Match and Tie-Breaker</p> <p>Rule 2: *.com – Match</p> <p>Rule 3: boat.fish.com – Not a Match</p> <p>FQDN: blue.boat.fish.com</p> <p>FQDN matches Rule 1 and Rule 2 (because the * matches one or more tokens). The firewall uses Rule 1 because it is the most specific.</p>
When working with wildcards (*) and Implicit-tail-match rules, there can be cases when the FQDN matches more	<p>Replace this:</p> <p>Rule: www.boat</p>

FQDN Comparison to DNS Proxy Rule	For Example
<p>than one rule and the tie-breaking algorithm weighs the rules equally.</p> <p>To avoid ambiguity, if rules with an Implicit-tail-match or a wildcard (*) can overlap, replace an Implicit-tail-match rule by specifying the tail token.</p>	<p>with this:</p> <p>Rule: www.boat.com</p>
Best Practices for Creating DNS Proxy Rules to Avoid Ambiguity and Unexpected Results	
<p>Include a top-level domain in the domain name to avoid invoking an Implicit-tail-match that may match the FQDN to more than one rule.</p>	boat.com
<p>If you use a wildcard (*), use it only as the leftmost token.</p> <p>This practice follows the common understanding of wildcard DNS records and the hierarchical nature of DNS.</p>	*.boat.com
<p>Use no more than one * in a rule.</p>	
<p>Use the * to establish a base rule associated with a DNS server, and use rules with more tokens to build exceptions to the rule, which you associate with different servers.</p> <p>The tie-breaking algorithm will select the most specific match, based on the number of matched tokens.</p>	<p>Rule: *.corporation.com – DNS server A</p> <p>Rule: www.corporation.com – DNS server B</p> <p>Rule: *.internal.corporation.com – DNS server C</p> <p>Rule: www.internal.corporation.com – DNS server D</p> <p>FQDN: mail.internal.corporation.com – matches DNS server C</p> <p>FQDN: mail.corporation.com – matches DNS server A</p>

DDNS

Learn about how Dynamic DNS (DDNS) service updates the mappings of domain names to IP addresses to provide accurate IP addresses to DNS clients.

- [Dynamic DNS Overview](#)
- [Configure Dynamic DNS for Firewall Interfaces](#)

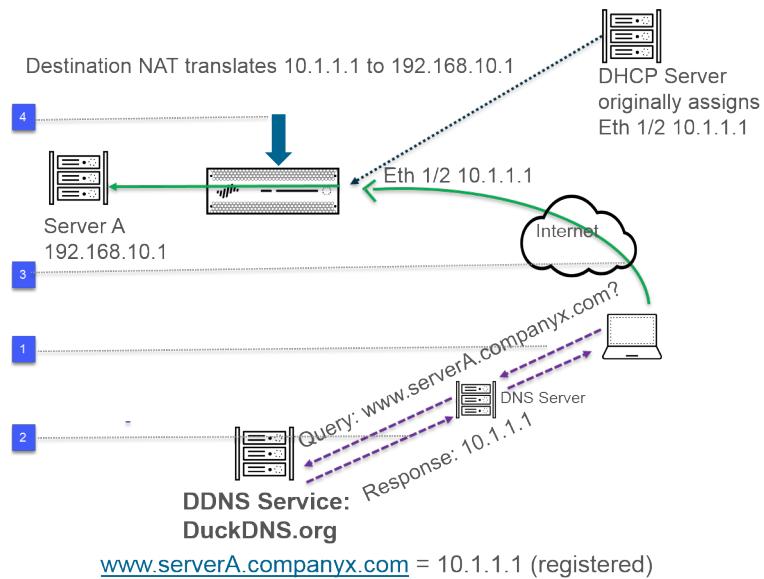
Dynamic DNS Overview

When you have services hosted behind the firewall and use destination NAT policies on the firewall to access those services or when you need to provide remote access to the firewall, you can register IPv4 address changes (whether the interface is a DHCP client receiving a dynamic address or has a static address) or IPv6 address changes (static address only) for the interface with a dynamic DNS (DDNS) service provider. The DDNS service automatically updates the domain name-to-IP address mappings to provide accurate IP addresses to DNS clients, which, in turn, can access the firewall and services behind the firewall. DDNS is often used in branch deployments that are hosting services. Without DDNS support for firewall interfaces, you would need external components to provide accurate IP addresses to clients.

The firewall supports the following **DDNS service providers**: DuckDNS, DynDNS, FreeDNS Afraid.org Dynamic API, FreeDNS Afraid.org, and No-IP. The individual DDNS service provider determines the services it provides, such as how many IP addresses it supports for a hostname and whether it supports IPv6 addresses. Palo Alto Networks® uses content updates to add new DDNS service providers and to provide updates to their services.

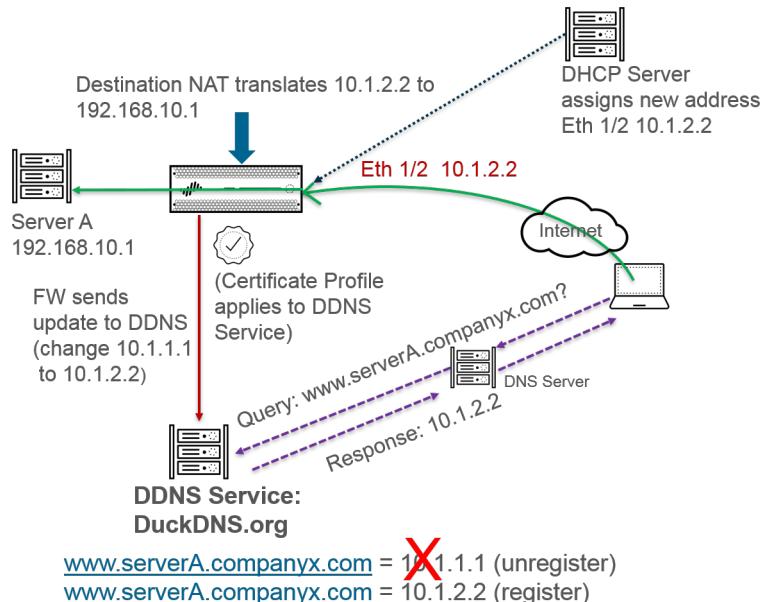
- For high availability (HA) configurations, make sure that content versions on the HA firewall peers (active/passive or active/active) are in sync because the firewall maintains the DDNS configuration based on the current Palo Alto Networks content release version. Palo Alto Networks can change or deprecate existing DDNS services through a content release. Additionally, a DDNS service provider can change the services it provides. A mismatch in content release versions between the HA peers can cause issues with their ability to use the DDNS service.
- The firewall does not support DDNS over an interface that is a Point-to-Point Protocol over Ethernet (PPPoE) termination point.

In the following example, the firewall is a DDNS client of a DDNS service provider. Initially, the DHCP server assigns IP address 10.1.1.1 to the Ethernet 1/2 interface. A destination NAT policy translates the public-facing 10.1.1.1 to the real address of Server A (192.168.10.1) behind the firewall.



1. When a user attempts to contact www.serverA.companyx.com, the user queries its local DNS server for the IP address. The www.serverA.companyx.com (set, for example, as a CNAME to your duckdns.org record: `serverA.companyx.duckdns.org`) is a domain belonging to the DDNS provider (DuckDNS in this example). The DNS server checks for the record with the DDNS provider to resolve the query.
2. The DNS server responds to the user with 10.1.1.1, which is the IP address for www.serverA.companyx.com.
3. The user packet with destination 10.1.1.1 goes to firewall interface Ethernet 1/2.
4. In this example, the firewall performs destination NAT and translates 10.1.1.1 to 192.168.10.2 before sending the packet to the destination.

After some time passes, DHCP assigns a new IP address to the firewall interface, which triggers a DDNS update, as follows:



1. The DHCP Server assigns a new IP address (10.1.2.2) to Ethernet 1/2.
2. When the firewall receives the new address, it sends an update to the DDNS service with the new address for www.serverA.companyx.com, which the DDNS service registers. (The firewall also sends regular updates based on the update interval you configure. The firewall sends DDNS updates over HTTPS port 443.)

Consequently, the next time the client queries the DNS server for the IP address of www.serverA.companyx.com and the DNS server checks the DDNS service, the DDNS service sends the updated address (10.1.2.2). Thus, the user successfully accesses a service or application through the firewall interface using the updated interface address.



If your firewall is configured for HA active/passive mode, be aware that the firewall sends DDNS updates to the DDNS service while the two HA firewall states are converging. After the HA states converge, DDNS is disabled on the passive firewall. For example, when two HA firewalls first boot up, they both send DDNS updates until they establish whether they are in HA active or passive mode. During this interval, you still see DDNS updates in system logs. After the HA states converge and each firewall notifies its clients that it is active or passive, the passive firewall no longer sends DDNS updates. (In HA active/active mode, each firewall has an independent DDNS configuration and doesn't synchronize the DDNS configuration.)

Configure Dynamic DNS for Firewall Interfaces

Before you configure [DDNS](#) for a firewall interface:

- Determine the hostname that you registered with your DDNS provider.
- Obtain the public SSL certificate from the DDNS service and import it in to the firewall.
- ([If you use FreeDNS Afraid.org v1 or FreeDNS Afraid.org Dynamic API v1](#)) On the DDNS server, the Dynamic DNS service tab includes the following option: **Link updates of the same IP together?** When this option is enabled, the DDNS service updates all hostnames in DNS records that contain the old IP address that is changing, not just the DNS record for a single hostname and IP address. To avoid updating DNS records of hosts you didn't intend to update, you should disable the **Link updates of the same IP together?** option so that the DDNS server updates only the DNS record that contains the specific hostname with the new IP address that is in the DDNS update.

STEP 1 | Configure DDNS.

1. Select **Network > Interfaces > Ethernet** and select a Layer 3 interface, subinterface, or Aggregate Ethernet (AE) interface; or select **Network > Interfaces > VLAN** and select an interface or subinterface.
2. Select **Advanced > DDNS** and select **Settings**.
3. **Enable DDNS.** You must initially enable DDNS to configure it. (If your DDNS configuration is unfinished, you can save it without enabling it so that you do not lose your partial configuration.)
4. Enter the **Update Interval (days)**, which is the number of days between updates that the firewall sends to the DDNS service to update IP addresses mapped to FQDNs (default is 1; range is 1 to 30). Choose an interval based on how frequently your IP addresses change. (The updates that the firewall sends at regular intervals are in addition to the updates the firewall sends upon receiving an address change. The updates sent at regular intervals are to ensure that updates sent per address change are not lost, for example.)
5. Enter the **Hostname** for the interface, which is already registered with the DDNS service (for example, www.serverA.companyx.com or serverA).

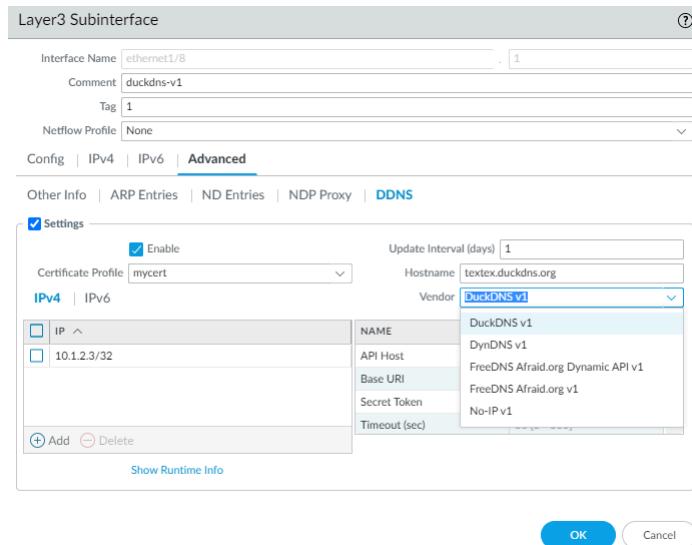


Make sure this hostname matches the hostname you registered with your DDNS service. You should enter an FQDN for the hostname; the firewall doesn't validate the hostname except to confirm that the syntax uses only valid characters allowed by DNS for a domain name.

6. Select **IPv4** and select one or more IPv4 addresses assigned to the interface or **Add** an IPv4 address to associate with the hostname (for example, 10.1.1.1). You can select only as many IPv4 addresses as the DDNS service allows. All selected IPv4 addresses are registered with the DDNS service. Select at least one IPv4 or one IPv6 address.
7. Select **IPv6** and select one or more IPv6 addresses assigned to the interface or **Add** an IPv6 address to associate with the hostname. You can select only as many IPv6 addresses as the DDNS service allows. All selected IPv6 addresses are registered with the DDNS service. Select at least one IPv4 or one IPv6 address.
8. Select or [create a new certificate profile \(Certificate Profile\)](#) using the imported SSL certificate from the DDNS service to verify the SSL certificate of the DDNS service

when the firewall first connects to a DDNS service to register an IP address and at every update. When the firewall connects to the DDNS service to send updates, the DDNS service presents the firewall with an SSL certificate signed by the certificate authority (CA) so that the firewall can authenticate the DDNS service.

- Select the **Vendor** (and version number) you are using for DDNS service.



Palo Alto Networks® may change the supported DDNS service providers via a content update.



In the Vendor field, the Palo Alto Network DDNS selection is a reserved DDNS service for Palo Alto Networks features such as SD-WAN and ZTP, and shouldn't be selected for this current task. If you mistakenly select Palo Alto Networks DDNS when the corresponding supporting feature isn't enabled, an error message will appear.

- The vendor choice determines the vendor-specific **Name** and **Value** fields below the Vendor field. Some Value fields are read-only to notify you of the parameters the firewall uses to connect to the DDNS service. Configure the remaining Value fields, such as a password that the DDNS service provides to you and a timeout that the firewall uses if it doesn't receive an update from the DDNS service.
- Click **OK**.

STEP 2 | **(Optional)** If you want the firewall to communicate with the DDNS service using an interface other than the management interface, configure a service route for DDNS ([Set Up Network Access for External Services](#)).

STEP 3 | Commit your changes.

STEP 4 | View DDNS information for the interface.

1. Select **Network > Interfaces > Ethernet** or **Network > Interfaces > VLAN** and select the interface you configured. (Interfaces with DDNS configured display the DDNS icon —  — in the Features field.)
2. Select **Advanced > DDNS** and **Settings**.
3. **Show Runtime Info** to see the DDNS information for the interface, including the Last return code (result of the last FQDN update) and Last time (date and time) the DDNS service received an FQDN update.

NAT

This section describes Network Address Translation (NAT) and how to configure the firewall for NAT. NAT allows you to translate private, non-routable IPv4 addresses to one or more globally-routable IPv4 addresses, thereby conserving an organization's routable IP addresses. NAT allows you to not disclose the real IP addresses of hosts that need access to public addresses and to manage traffic by performing port forwarding. You can use NAT to solve network design challenges, enabling networks with identical IP subnets to communicate with each other. The firewall supports NAT on Layer 3 and virtual wire interfaces.

The [NAT64](#) option translates between IPv6 and IPv4 addresses, providing connectivity between networks using disparate IP addressing schemes, and therefore a migration path to IPv6 addressing. IPv6-to-IPv6 Network Prefix Translation ([NPTv6](#)) translates one IPv6 prefix to another IPv6 prefix. PAN-OS supports all of these functions.

If you use private IP addresses within your internal networks, you must use NAT to translate the private addresses to public addresses that can be routed on external networks. In PAN-OS, you create NAT policy rules that instruct the firewall which packet addresses and ports need translation and what the translated addresses and ports are.

- [NAT Policy Rules](#)
- [Source NAT and Destination NAT](#)
- [Destination NAT with DNS Rewrite Use Cases](#)
- [NAT Rule Capacities](#)
- [Dynamic IP and Port NAT Oversubscription](#)
- [Dataplane NAT Memory Statistics](#)
- [Configure NAT](#)
- [NAT Configuration Examples](#)

NAT Policy Rules

- [NAT Policy Overview](#)
- [NAT Address Pools Identified as Address Objects](#)
- [Proxy ARP for NAT Address Pools](#)

NAT Policy Overview

You configure a NAT rule to match a packet's source zone and destination zone, at a minimum. In addition to zones, you can configure matching criteria based on the packet's destination interface, source and destination address, and service. You can configure multiple NAT rules. The firewall evaluates the rules in order from the top down. Once a packet matches the criteria of a single NAT rule, the packet is not subjected to additional NAT rules. Therefore, your list of NAT rules should be in order from most specific to least specific so that packets are subjected to the most specific rule you created for them.

It is important to understand that in firewall policy rules, the set of IPv4 addresses is treated as a subset of the set of IPv6 addresses. However, the set of IPv6 addresses is not a subset of the set of IPv4 addresses. An IPv4 address can match a set or range of IPv6 addresses; but an IPv6 address cannot match a set or range of IPv4 addresses.

In all policy types, the keyword **any** for a source or destination address means any IPv4 or IPv6 address. The keyword **any** is equivalent to `::/0`. If you want to express "any IPv4 address", specify `0.0.0.0/0`.

During policy matching, the firewall converts an IPv4 address into an IPv6 prefix where the first 96 bits are 0. An address of `::/8` means, match the rule if the first 8 bits are 0. All IPv4 addresses will match `::/8`, `::/9`, `::/10`, `::/11`, ... `::/16`, ... `::/32`, ... through `::/96`.

If you want to express "any IPv6 address, but no IPv4 addresses", you must configure two rules. The first rule denies `0.0.0.0/0` to deny any IPv4 address (as the source or destination address), and the second rule has `::/0` to mean any IPv6 address (as the source or destination address), to satisfy your requirement.

Static NAT rules do not have precedence over other forms of NAT. Therefore, for static NAT to work, the static NAT rules must be above all other NAT rules in the list on the firewall.

NAT rules provide address translation, and are different from security policy rules, which allow or deny packets. It is important to understand the firewall's flow logic when it applies NAT rules and security policy rules so that you can determine what rules you need, based on the zones you have defined. You must configure security policy rules to allow the NAT traffic.

Upon ingress, the firewall inspects the packet and does a route lookup to determine the egress interface and zone. Then the firewall determines if the packet matches one of the NAT rules that have been defined, based on source and/or destination zone. It then evaluates and applies any security policies that match the packet based on the original (pre-NAT) source and destination addresses, but the post-NAT zones. Finally, upon egress, for a matching NAT rule, the firewall translates the source and/or destination address and port numbers.

Keep in mind that the translation of the IP address and port do not occur until the packet leaves the firewall. The NAT rules and security policies apply to the original IP address (the pre-NAT address). A NAT rule is configured based on the zone associated with a pre-NAT IP address.

Security policies differ from NAT rules because security policies examine post-NAT zones to determine whether the packet is allowed or not. Because the very nature of NAT is to modify source or destination IP addresses, which can result in modifying the packet's outgoing interface and zone, security policies are enforced on the post-NAT zone.



A SIP call sometimes experiences one-way audio when going through the firewall because the call manager sends a SIP message on behalf of the phone to set up the connection. When the message from the call manager reaches the firewall, the SIP ALG must put the IP address of the phone through NAT. If the call manager and the phones are not in the same security zone, the NAT lookup of the IP address of the phone is done using the call manager zone. The NAT policy should take this into consideration.

No-NAT rules are configured to allow exclusion of IP addresses defined within the range of NAT rules defined later in the NAT policy. To define a no-NAT policy, specify all of the match criteria and select No Source Translation in the source translation column.

You can verify the NAT rules processed by selecting **Device > Troubleshooting** and testing the traffic matches for the NAT rule. For example:

Test Configuration	Test Result	Result Detail				
<p>Select Test: NAT Policy Match</p> <p>From: I3-vlan-trust</p> <p>To: I3-untrust</p> <p>Source: 10.54.21.28</p> <p>Destination: 8.8.8.8</p> <p>Source Port: [1 - 65535]</p> <p>Destination Port: 445</p> <p>Protocol: 6</p> <p>To Interface: None</p> <p>Ha Device ID: [0 - 1]</p> <p>Execute Reset</p>	NAT Policy Match Result	<table border="1"> <thead> <tr> <th>Name</th><th>Value</th></tr> </thead> <tbody> <tr> <td>Result</td><td>access-corp</td></tr> </tbody> </table>	Name	Value	Result	access-corp
Name	Value					
Result	access-corp					

NAT Address Pools Identified as Address Objects

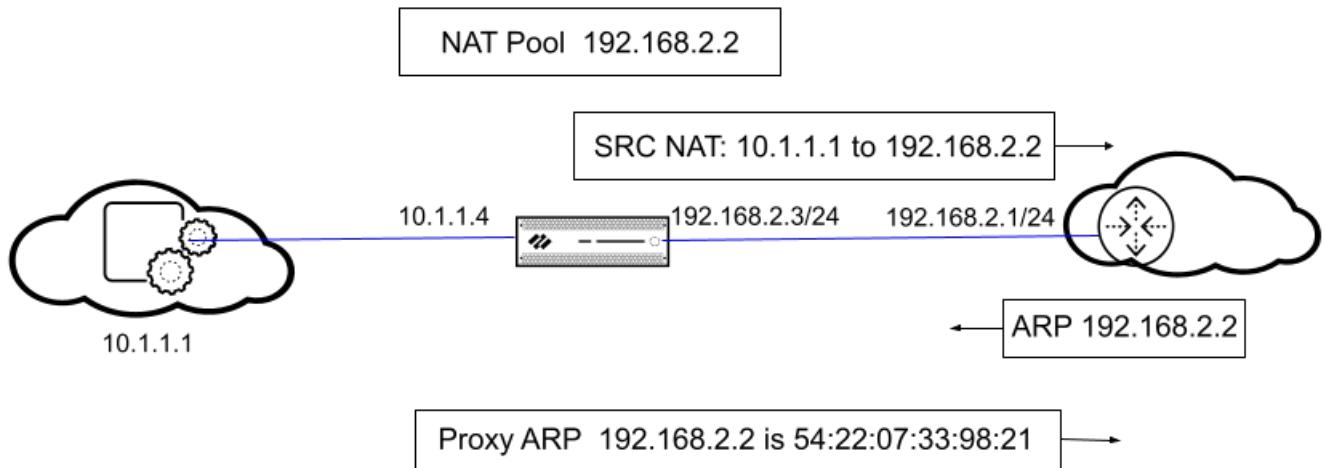
When configuring a **Dynamic IP** or **Dynamic IP and Port** NAT address pool in a NAT policy rule, it is typical to configure the pool of translated addresses with address objects. Each address object can be a host IP address, IP address range, or IP subnet.



Because both NAT rules and security policy rules use address objects, it is a best practice to distinguish between them by naming an address object used for NAT with a prefix, such as "NAT-name."

Proxy ARP for NAT Address Pools

NAT address pools are not bound to any interfaces. The following figure illustrates the behavior of the firewall when it is performing proxy ARP for an address in a NAT address pool.



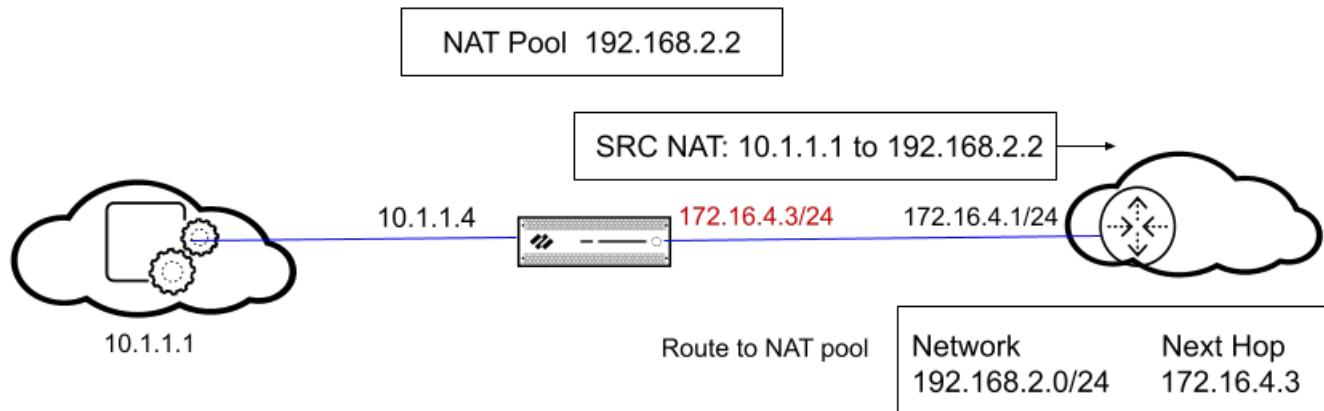
The firewall performs source NAT for a client, translating the source address 10.1.1.1 to the address in the NAT pool, 192.168.2.2. The translated packet is sent on to a router.

For the return traffic, the router doesn't know how to reach 192.168.2.2 (because that IP address is just an address in the NAT address pool), so it sends an ARP request packet to the firewall.

In our first scenario, when the NAT pool address (192.168.2.2) is in the same subnet as the egress/ingress interface IP address (192.168.2.3/24), the firewall can send a proxy ARP reply to the router, indicating the Layer 2 MAC address for 192.168.2.2 is 54:22:07:33:98:21, as shown in the figure above.

No Proxy ARP When the NAT Pool Address Isn't a Subnet of the Egress/Ingress Interface

In our second scenario, the NAT pool address (192.168.2.2) isn't a subnet of an interface on the firewall, so the firewall won't send a proxy ARP reply to the router. This means that the router must be configured with the necessary route to know where to send packets destined for 192.168.2.2, in order to ensure the return traffic is routed back to the firewall, as shown in the figure below.



Source NAT and Destination NAT

The firewall supports both source address and/or port translation and destination address and/or port translation.

- [Source NAT](#)
- [Destination NAT](#)

Source NAT

Source NAT is typically used by internal users to access the Internet; the source address is translated and thereby kept private. There are three types of source NAT:

- **Dynamic IP and Port (DIPP)**—Allows multiple hosts to have their source IP addresses translated to the same public IP address with different port numbers. The dynamic translation is to the next available address in the NAT address pool, which you configure as a **Translated Address** pool to be an IP address, range of addresses, a subnet, or a combination of these.

As an alternative to using the next address in the NAT address pool, DIPP allows you to specify the address of the **Interface** itself. The advantage of specifying the interface in the NAT rule is that the NAT rule will be automatically updated to use any address subsequently acquired by the interface. DIPP is sometimes referred to as interface-based NAT or network address port translation (NAPT).

DIPP has a default NAT oversubscription rate, which is the number of times that the same translated IP address and port pair can be used concurrently. For more information, see [Dynamic IP and Port NAT Oversubscription](#) and [Modify the Oversubscription Rate for DIPP NAT](#).



(*Affects only PA-7000 Series firewalls that do not use second-generation PA-7050-SMC-B or PA-7080-SMC-B Switch Management Cards*) When you use Point-to-Point Tunnel Protocol (PPTP) with DIPP NAT, the firewall is limited to using a translated IP address-and-port pair for only one connection; the firewall does not support DIPP NAT. The workaround is to upgrade the PA-7000 Series firewall to a second-generation SMC-B card.

- **Dynamic IP**—Allows the one-to-one, dynamic translation of a source IP address only (no port number) to the next available address in the NAT address pool. The size of the NAT pool should be equal to the number of internal hosts that require address translations. By default, if the source address pool is larger than the NAT address pool and eventually all of the NAT addresses are allocated, new connections that need address translation are dropped. To override this default behavior, use **Advanced (Dynamic IP/Port Fallback)** to enable use of DIPP addresses when necessary. In either event, as sessions terminate and the addresses in the pool become available, they can be allocated to translate new connections.

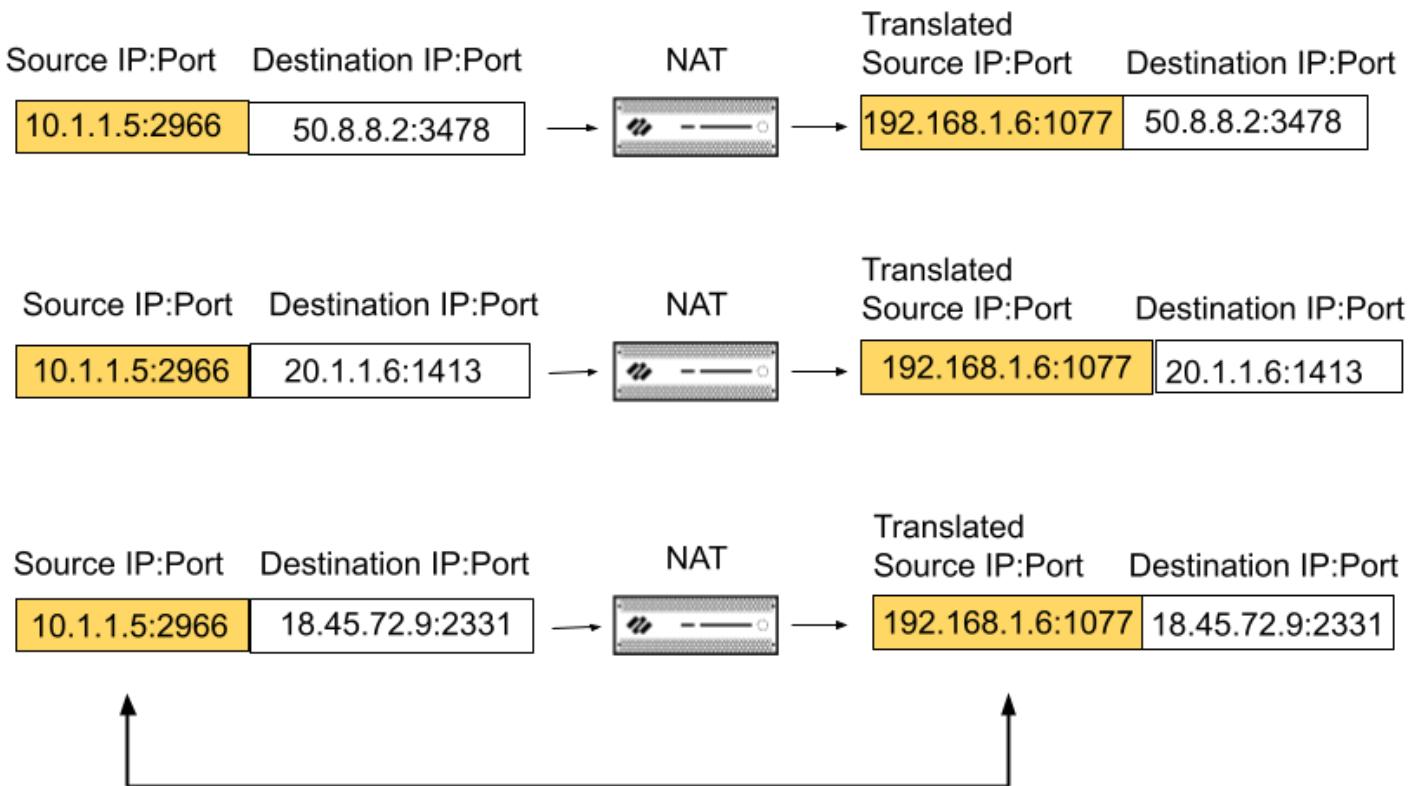
Dynamic IP NAT supports the option for you to [Reserve Dynamic IP NAT Addresses](#).

- **Static IP**—Allows the 1-to-1, static translation of a source IP address, but leaves the source port unchanged. A common scenario for a static IP translation is an internal server that must be available to the Internet.

Beginning with PAN-OS 10.2.4, [persistent NAT for DIPP](#) is available on all firewalls. VoIP, video, cloud-based video conferencing, audio conferencing, and other applications often use

DIPP and may require the Session Traversal Utilities for NAT (STUN) protocol. DIPP NAT uses symmetric NAT, which may have compatibility issues with applications that use STUN. To alleviate these issues, persistent NAT for DIPP provides additional support for connectivity with such applications.

When persistent NAT for DIPP is enabled, the binding of a private source IP address/port pair to a specific public (translated) source IP address/port pair persists for subsequent sessions that arrive having that same original source IP address/port pair. The following example shows three sessions:



The NAT DIPP binding is persistent for subsequent sessions that have the same Source IP:Port

In this example, original source IP address/port 10.1.1.5:2966 is bound to the translated source IP address/port 192.168.1.6:1077 in Session 1. That binding is persistent in Session 2 and Session 3, which have the same original source IP address/port, but different destination addresses. The persistence of the binding ends after all of the sessions for that source IP address/port pair have ended.

In Session 1 of the example, the Destination port is 3478, the default STUN port.

When persistent NAT for DIPP is enabled, it applies to all NAT and [NAT64](#) rules; it is a global setting. Management plane or dataplane logs will indicate NAT DIPP/STUN support has been enabled.

The persistent NAT for DIPP setting (enabled or disabled) survives across firewall reboots.

Destination NAT

Destination NAT is performed on incoming packets when the firewall translates a destination address to a different destination address; for example, it translates a public destination address to a private destination address. Destination NAT also offers the option to perform port forwarding or port translation.

Destination NAT allows static and dynamic translation:

- **Static IP**—You can configure a one-to-one, [static translation](#) in several formats. You can specify that the original packet have a single destination IP address, a range of IP addresses, or an IP netmask, as long as the translated packet is in the same format and specifies the same number of IP addresses. The firewall statically translates an original destination address to the same translated destination address each time. That is, if there is more than one destination address, the firewall translates the first destination address configured for the original packet to the first destination address configured for the translated packet, and translates the second original destination address configured to the second translated destination address configured, and so on, always using the same translation.

If you use destination NAT to translate a static IPv4 address, you might also use DNS services on one side of the firewall to resolve FQDNs for a client on the other side. When the DNS response containing the IPv4 address traverses the firewall, the DNS server provides an internal IP address to an external device, or vice versa. Beginning with PAN-OS 9.0.2 and in later 9.0 releases, you can configure the firewall to rewrite the IP address in the DNS response (that matches the rule) so that the client receives the appropriate address to reach the destination service. The applicable [DNS rewrite use case](#) determines how you configure such a rewrite.

- **Dynamic IP (with session distribution)**—Destination NAT allows you to translate the original destination address to a destination host or server that has a [dynamic IP address](#), meaning an address object that uses an FQDN, which can return multiple addresses from DNS. Dynamic IP (with session distribution) supports IPv4 addresses only. Destination NAT using a dynamic IP address is especially helpful in cloud deployments that use dynamic IP addressing.



*When you configure destination NAT for an original (pre-NAT) destination address that is an FQDN or dynamic address group (DAG), the translation type must be **Dynamic IP (with session distribution)**, not Static IP.*

If the translated destination address resolves to more than one address, the firewall distributes incoming NAT sessions among the multiple addresses to provide improved session distribution. Distribution is based on one of several methods: round-robin (the default method), source

IP hash, IP modulo, IP hash, or least sessions. If a DNS server returns more than 32 IPv4 addresses for an FQDN, the firewall uses the first 32 addresses in the packet.



If the translated address is an address object of type FQDN that resolves to only IPv6 addresses, the destination NAT policy rule considers the FQDN as unresolved.

Using **Dynamic IP (with session distribution)** allows you to translate multiple pre-NAT destination IP addresses M to multiple post-NAT destination IP addresses N. A many-to-many translation means there can be M x N destination NAT translations using a single NAT rule.

You must select the Destination Address Translation Type (**Policy > NAT > Translated Packet**) as **Dynamic IP (with session distribution)**, when you configure the pre-NAT address as FQDN object in **Policy > NAT > Original Packet**.



For destination NAT, the best practice is to:

- Use **Static IP** address translation for static IP addresses, which allows the firewall to check and ensure that the number of original destination IP addresses equals the number of translated destination IP addresses.
- Use **Dynamic IP (with session distribution)** address translation only for FQDN-based dynamic addresses (the firewall does not perform an IP address number check).

The following are common examples of destination NAT translations that the firewall allows:

Translation Type	Original Packet's Destination Address	Maps to Translated Packet's Destination Address	Notes
Static IP	192.168.1.1	2.2.2.2	Original packet and translated packet each have one possible destination address.
	192.168.1.1-192.168.1.4	2.2.2.1-2.2.2.4	Original packet and translated packet each have four possible destination addresses: 192.168.1.1 always maps to 2.2.2.1 192.168.1.2 always maps to 2.2.2.2 192.168.1.3 always maps to 2.2.2.3 192.168.1.4 always maps to 2.2.2.4

Translation Type	Original Packet's Destination Address	Maps to Translated Packet's Destination Address	Notes
	192.168.1.1/30	2.2.2.1/30	<p>Original packet and translated packet each have four possible destination addresses:</p> <ul style="list-style-type: none"> 192.168.1.1 always maps to 2.2.2.1 192.168.1.2 always maps to 2.2.2.2 192.168.1.3 always maps to 2.2.2.3 192.168.1.4 always maps to 2.2.2.4
Dynamic IP (with session distribution)	192.168.1.1/30	domainname.com	Original packet has four destination addresses and if, for example, the FQDN in the translated destination address resolves to five IP addresses, then there are 20 possible destination NAT translations in a single NAT rule.

One common use for destination NAT is to configure several NAT rules that map a single public destination address to several private destination host addresses assigned to servers or services. In this case, the destination port numbers are used to identify the destination hosts. For example:

- **Port Forwarding**—Can translate a public destination address and port number to a private destination address but keeps the same port number.
- **Port Translation**—Can translate a public destination address and port number to a private destination address and a different port number, thus keeping the actual port number private. The port translation is configured by entering a **Translated Port** on the **Translated Packet** tab in the NAT policy rule. See the [Destination NAT with Port Translation Example](#).

Destination NAT with DNS Rewrite Use Cases

When you use destination NAT to perform a static translation from one IPv4 address to a different IPv4 address, you may also be using DNS services on one side of the firewall to resolve FQDNs for a client. When the DNS response containing the IP address traverses the firewall to go to the client, the firewall doesn't perform NAT on that IP address, so the DNS server provides an internal IP address to an external device, or vice versa, resulting in the DNS client being unable to connect to the destination service.

To avoid that problem, you can [configure the firewall to rewrite the IP address in the DNS response](#) (from the A Record) based on the translated IP address configured for the NAT policy rule. The firewall performs NAT on the IPv4 address (the FQDN resolution) in the DNS response before forwarding the response to the client; thus, the client receives the appropriate address to reach the destination service. A single NAT policy rule causes the firewall to perform NAT on packets that match the rule, and also causes the firewall to perform NAT on IP addresses in DNS responses that match the original destination address or translated destination address in the rule.

DNS rewrite occurs at the global level; the firewall maps the Destination Address on the Original Packet tab to the Destination Address on the Translated Packet tab. All other fields on the Original Packet tab are ignored. When a DNS response packet arrives, the firewall checks whether the response contains any A Record that matches one of the mapped destination addresses, based on the direction, as follows.

You must specify how the firewall performs NAT on the IP address in the DNS response relative to the NAT rule: **reverse** or **forward**:

- **reverse**—If the DNS response matches the **Translated** Destination Address in the rule, translate the DNS response using the reverse translation that the rule uses. For example, if the rule translates IP address **1.1.1.10 to 192.168.1.10**, the firewall rewrites a DNS response of **192.168.1.10 to 1.1.1.10**.
- **forward**—If the DNS response matches the **Original** Destination Address in the rule, translate the DNS response using the same translation the rule uses. For example, if the rule translates IP address **1.1.1.10 to 192.168.1.10**, the firewall rewrites a DNS response of **1.1.1.10 to 192.168.1.10**.



If you have an overlapping NAT rule with DNS Rewrite disabled, and a NAT rule below it that has DNS Rewrite enabled and is included in the overlap, the firewall rewrites the DNS response according to the overlapped NAT rule (in either **reverse** or **forward** setting). The rewrite takes precedence and the order of the NAT rules is ignored.

Consider the use cases for configuring DNS rewrite:

- [Destination NAT with DNS Rewrite Reverse Use Cases](#)
- [Destination NAT with DNS Rewrite Forward Use Cases](#)

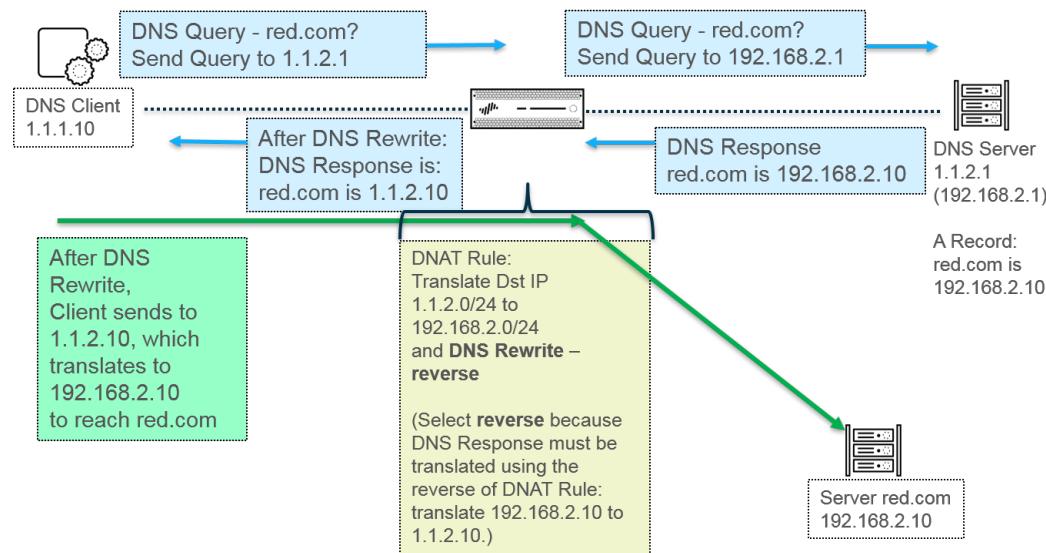
Destination NAT with DNS Rewrite Reverse Use Cases

The following use cases illustrate [destination NAT with DNS rewrite](#) enabled in the **reverse** direction. The difference between these two use cases is simply whether the DNS client, DNS server, and destination server are on the public or internal side of the firewall. In either case, the DNS client is on the opposite side of the firewall from its ultimate destination server. (If your DNS client and its ultimate destination server are on the same side of the firewall, consider [Destination NAT with DNS Rewrite Forward Use Cases](#) 3 and 4.)

Use case 1 illustrates the DNS client on the public side of the firewall, while the DNS server and the ultimate destination server are both on the internal side. This case requires DNS rewrite in the reverse direction. The DNS client queries for the IP address of red.com. Based on the NAT rule, the firewall translates the query (originally going to public address 1.1.2.1) to internal address 192.168.2.1. The DNS server responds that red.com has IP address 192.168.2.10. The rule includes **Enable DNS Rewrite - reverse** and the DNS response of 192.168.2.10 matches the destination Translated Address of 192.168.2.0/24 in the rule, so the firewall translates the DNS

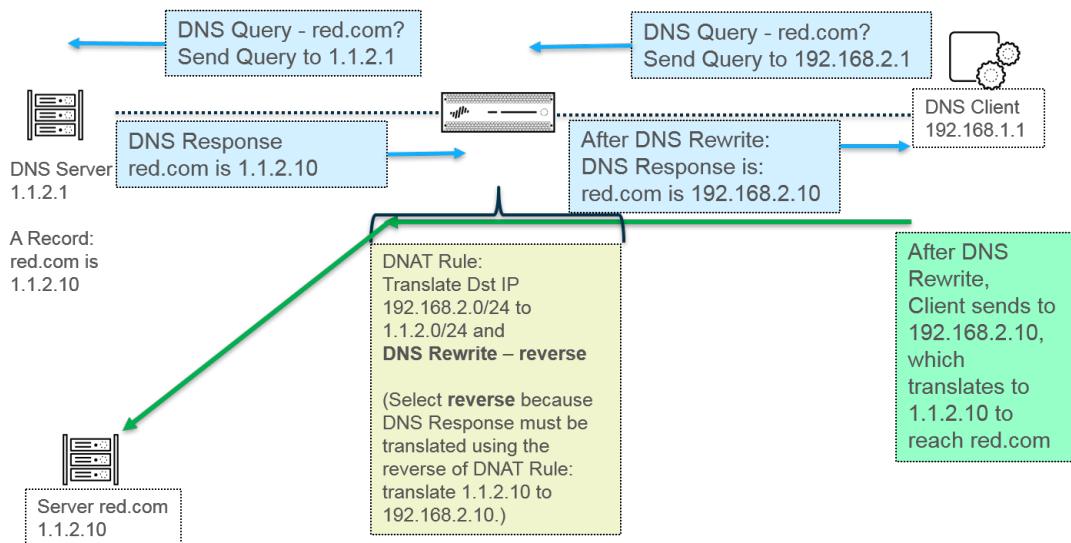
response using the **reverse** translation that the rule uses. The rule says translate 1.1.2.0/24 to 192.168.2.0/24, so the firewall rewrites the DNS response of 192.168.2.10 to 1.1.2.10. The DNS client receives the response and sends to 1.1.2.10, which the rule translates to 192.168.2.10 to reach server red.com.

Use case 1 summary: DNS client and destination server are on opposite sides of the firewall. The DNS server provides an address that matches the translated destination address in the NAT rule, so translate the DNS response using the **reverse** translation of the NAT rule.



Use case 2 illustrates the DNS client on the internal side of the firewall, while the DNS server and the ultimate destination server are both on the public side. This case requires DNS rewrite in the reverse direction. The DNS client queries for the IP address of red.com. Based on the NAT rule, the firewall translates the query (originally going to internal address 192.168.2.1) to the public address 1.1.2.1. The DNS server responds that red.com has IP address 1.1.2.10. The rule includes **Enable DNS Rewrite - reverse** and the DNS response of 1.1.2.10 matches the destination Translated Address of 1.1.2.0/24 in the rule, so the firewall translates the DNS response using the **reverse** translation that the rule uses. The rule says translate 192.168.2.0/24 to 1.1.2.0/24, so the firewall rewrites the DNS response 1.1.2.10 to 192.168.2.10. The DNS client receives the response and sends to 192.168.2.10, which the rule translates to 1.1.2.10 to reach server red.com.

Use case 2 summary is the same as Use case 1 summary: DNS client and destination server are on opposite sides of the firewall. The DNS server provides an address that matches the translated destination address in the NAT rule, so translate the DNS response using the **reverse** translation of the NAT rule.



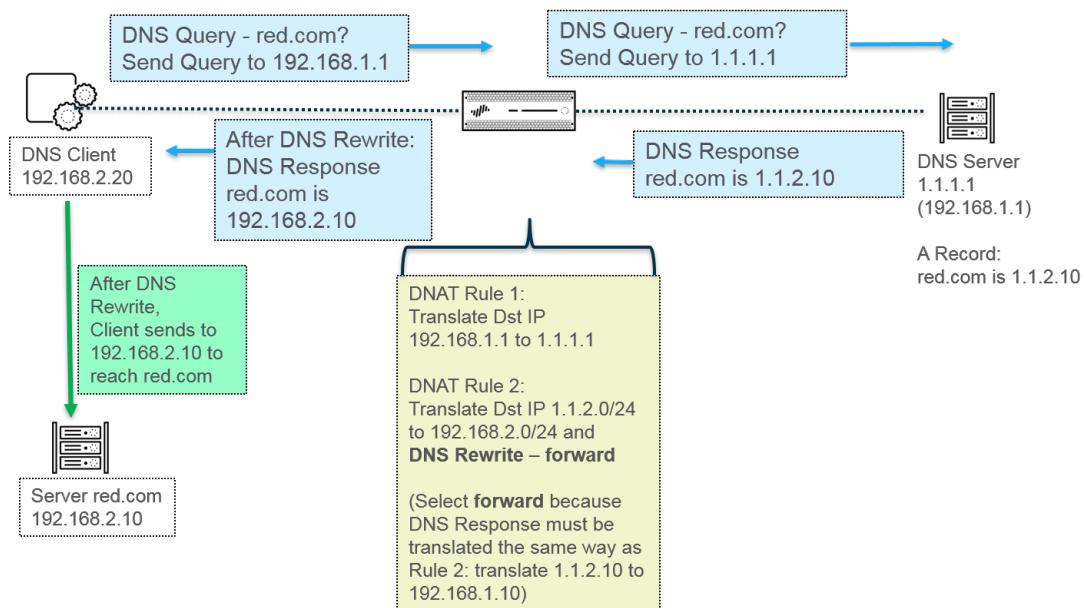
To implement DNS rewrite, [Configure Destination NAT with DNS Rewrite](#).

Destination NAT with DNS Rewrite Forward Use Cases

The following use cases illustrate [destination NAT with DNS rewrite](#) enabled in the **forward** direction. The difference between these two use cases is simply whether the DNS client, DNS server, and destination server are on the public or internal side of the firewall. In either case, the DNS client is on the same side of the firewall as its ultimate destination server. (If your DNS client and its ultimate destination server are on opposite sides of the firewall, consider [Destination NAT with DNS Rewrite Reverse Use Cases 1 and 2](#).)

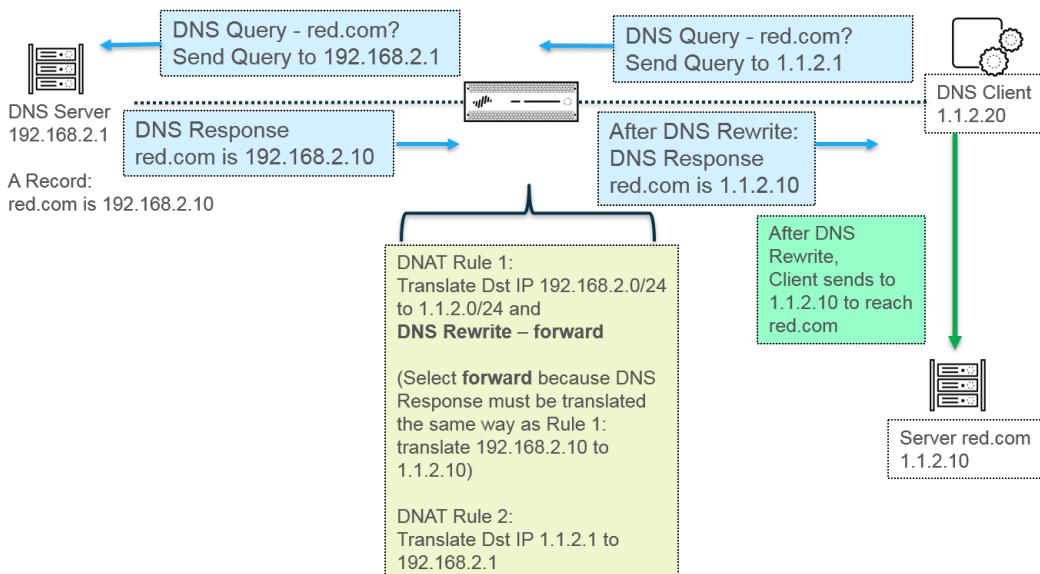
Use case 3 illustrates the DNS client and the ultimate destination server both on the internal side of the firewall, while the DNS server is on the public side. This case requires DNS rewrite in the forward direction. The DNS client queries for the IP address of red.com. Based on Rule 1, the firewall translates the query (originally going to internal address 192.168.1.1) to 1.1.1.1. The DNS server responds that red.com has IP address 1.1.2.10. Rule 2 includes [Enable DNS Rewrite - forward](#) and the DNS response of 1.1.2.10 matches the original destination address of 1.1.2.0/24 in Rule 2, so the firewall translates the DNS response using the **same** translation the rule uses. Rule 2 says translate 1.1.2.0/24 to 192.168.2.0/24, so the firewall rewrites DNS response 1.1.2.10 to 192.168.2.10. The DNS client receives the response and sends to 192.168.2.10 to reach server red.com.

Use case 3 summary: DNS client and destination server are on the same side of the firewall. The DNS server provides an address that matches the original destination address in the NAT rule, so translate the DNS response using the same (**forward**) translation as the NAT rule.



Use case 4 illustrates the DNS client and the ultimate destination server both on the public side of the firewall, while the DNS server is on the internal side. This case requires DNS Rewrite in the forward direction. The DNS client queries for the IP address of red.com. Based on Rule 2, the firewall translates the query (originally going to public destination 1.1.2.1) to 192.168.2.1. The DNS server responds that red.com has IP address 192.168.2.10. Rule 1 includes **Enable DNS Rewrite - forward** and the DNS response of 192.168.2.10 matches the original destination address of 192.168.2.0/24 in Rule 1, so the firewall translates the DNS response using the **same** translation the rule uses. Rule 1 says translate 192.168.2.0/24 to 1.1.2.0/24, so the firewall rewrites DNS response 192.168.2.10 to 1.1.2.10. The DNS client receives the response and sends to 1.1.2.10 to reach server red.com.

Use case 4 summary is the same as Use case 3 summary: DNS client and destination server are on the same side of the firewall. The DNS server provides an address that matches the original destination address in the NAT rule, so translate the DNS response using the same (**forward**) translation as the NAT rule.



To implement DNS rewrite, [Configure Destination NAT with DNS Rewrite](#).

NAT Rule Capacities

The number of NAT rules allowed is based on the firewall model. Individual rule limits are set for static, Dynamic IP (DIP), and Dynamic IP and Port (DIPP) NAT. The sum of the number of rules used for these NAT types cannot exceed the total NAT rule capacity. For DIPP, the rule limit is based on the oversubscription setting (8, 4, 2, or 1) of the firewall and the assumption of one translated IP address per rule. To see model-specific NAT rule limits and translated IP address limits, use the [Compare Firewalls](#) tool.

Consider the following when working with NAT rules:

- If you run out of pool resources, you cannot create more NAT rules, even if the model's maximum rule count has not been reached.
- If you consolidate NAT rules, the logging and reporting will also be consolidated. The statistics are provided per the rule, not per all of the addresses within the rule. If you need granular logging and reporting, do not combine the rules.

Dynamic IP and Port NAT Oversubscription

Dynamic IP and Port (DIPP) NAT allows you to use each translated IP address and port pair multiple times (8, 4, or 2 times) in concurrent sessions. This reusability of an IP address and port (known as oversubscription) provides scalability for customers who have too few public IP addresses. The design is based on the assumption that hosts are connecting to different destinations, therefore sessions can be uniquely identified and collisions are unlikely. The oversubscription rate in effect multiplies the original size of the address/port pool to 8, 4, or 2 times the size. For example, the default limit of 64K concurrent sessions allowed, when multiplied by an oversubscription rate of 8, results in 512K concurrent sessions allowed.

The oversubscription rates that are allowed vary based on the model. The oversubscription rate is global; it applies to the firewall. This oversubscription rate is set by default and consumes memory, even if you have enough public IP addresses available to make oversubscription unnecessary. You can reduce the rate from the default setting to a lower setting or even 1 (which means no oversubscription). By configuring a reduced rate, you decrease the number of source device translations possible, but increase the DIP and DIPP NAT rule capacities. To change the default rate, see [Modify the Oversubscription Rate for DIPP NAT](#).

If you select **Platform Default**, your explicit configuration of oversubscription is turned off and the NAT default DIPP pool oversubscription rate for the model applies, as shown in the [Product Selection tool](#). The **Platform Default** setting allows for an upgrade or downgrade of a software release.

The firewall supports a maximum of 256 translated IP addresses per NAT rule, and each model supports a maximum number of translated IP addresses (for all NAT rules combined). If oversubscription causes the maximum translated addresses per rule (256) to be exceeded, the firewall will automatically reduce the oversubscription ratio in an effort to have the commit succeed. However, if your NAT rules result in translations that exceed the maximum translated addresses for the model, the commit will fail.

Dataplane NAT Memory Statistics

The **show running global-ippool** command displays statistics related to NAT memory consumption for a pool. The Size column displays the number of bytes of memory that the resource pool is using. The Ratio column displays the oversubscription ratio (for DIPP pools only). The lines of pool and memory statistics are explained in the following sample output:

```
admin@PA-7050-HA-0 (active-primary)>show running global-ippool
```

Idx	Type	From	To	Num	Ref.Cnt	Size	Ratio
1	Dynamic IP	201.0.0.0-201.0.255.255	210.0.0.0	4096	2	657072	N/A
2	Dynamic IP	202.0.0.0-202.0.0.255	220.0.0.0	256	1	41232	N/A
3	Dynamic IP/Port	200.0.2.100-200.0.2.100	200.0.3.11	1	1	68720	8

Usable NAT DIP/DIPP shared memory size: 58490064 ← Total physical NAT memory (bytes)

Used NAT DIP/DIPP shared memory size: 767024 (1.3%) ← Bytes and % of usable NAT memory

DynamicIP NAT Pool: 2 (1.19%) ← Number of DIP pools in use and % of total usable memory that all DIP pools use

DynamicIP/Port NAT Pool: 1 (0.12%) ← Number of DIPP pools in use and % of total usable memory that all DIPP pools use

For NAT pool statistics for a virtual system, the **show running ippool** command has columns indicating the memory size used per NAT rule and the oversubscription ratio used (for DIPP rules). The following is sample output for the command.

```
admin@PA-7050-HA-0 vsys1 (active-primary)> show running ippool
```

VSYS 1 has 4 NAT rules, DIP and DIPP rules:					
Rule	Type	Used	Available	Mem Size	Ratio
nat1	Dynamic IP	0	4096	788144	0
nat2	Dynamic IP	0	256	49424	0
nat3	Dynamic IP/Port	0	638976	100976	4
nat11	Dynamic IP	0	4096	788144	0

A field in the output of the **show running nat-rule-ippool rule** command shows the memory (bytes) used per NAT rule. The following is sample output for the command, with the memory usage for the rule encircled.

```
admin@PA-7050-HA-0 (active-primary)>show running nat-rule-ippool rule nat1
```

VSYS 1 Rule nat1:
Rule:nat1, Pool index: 1, memory usage: 788144

Reserve IP: no
201.0.0.0-201.0.255.255 =>
210.0.0.0-210.0.15.255

Source Xlat-Source Ref.Cnt (F) TTL(s)

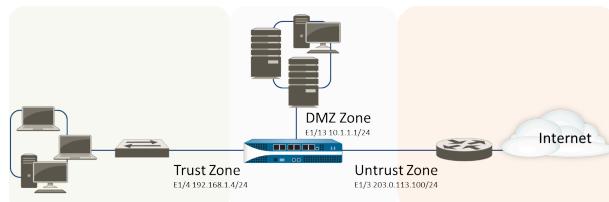
Total IPs in use: 0
Total entries in time-reserve cache: 0
Total freelist left: 4096

Configure NAT

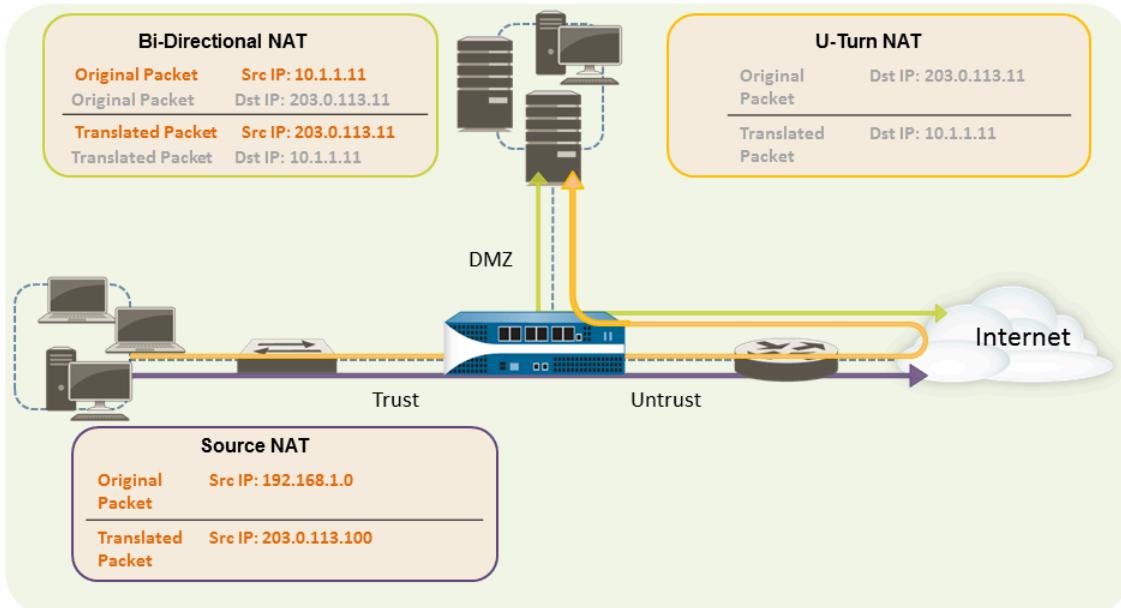
Perform the following tasks to configure various aspects of NAT. In addition to the examples below, there are examples in the section [NAT Configuration Examples](#).

- Translate Internal Client IP Addresses to Your Public IP Address (Source DIPP NAT)
- Enable Clients on the Internal Network to Access your Public Servers (Destination U-Turn NAT)
- Enable Bi-Directional Address Translation for Your Public-Facing Servers (Static Source NAT)
- Configure Destination NAT with DNS Rewrite
- Configure Destination NAT Using Dynamic IP Addresses
- Modify the Oversubscription Rate for DIPP NAT
- Reserve Dynamic IP NAT Addresses
- Disable NAT for a Specific Host or Interface

The first three NAT examples in this section are based on the following topology:



Based on this topology, there are three NAT policies we need to create as follows:



- To enable the clients on the internal network to access resources on the Internet, the internal 192.168.1.0 addresses will need to be translated to publicly routable addresses. In this case, we will configure source NAT (the purple enclosure and arrow above), using the egress interface address, 203.0.113.100, as the source address in all packets that leave the firewall

from the internal zone. See [Translate Internal Client IP Addresses to Your Public IP Address \(Source DIPP NAT\)](#) for instructions.

- To enable clients on the internal network to access the public web server in the DMZ zone, we must configure a NAT rule that redirects the packet from the external network, where the original routing table lookup will determine it should go based on the destination address of 203.0.113.11 within the packet, to the actual address of the web server on the DMZ network of 10.1.1.11. To do this you must create a NAT rule from the trust zone (where the source address in the packet is) to the untrust zone (where the original destination address is) to translate the destination address to an address in the DMZ zone. This type of destination NAT is called *U-Turn NAT* (the yellow enclosure and arrow above). See [Enable Clients on the Internal Network to Access your Public Servers \(Destination U-Turn NAT\)](#) for instructions.
- To enable the web server—which has both a private IP address on the DMZ network and a public-facing address for access by external users—to both send and receive requests, the firewall must translate the incoming packets from the public IP address to the private IP address and the outgoing packets from the private IP address to the public IP address. On the firewall, you can accomplish this with a single bi-directional static source NAT policy (the green enclosure and arrow above). See [Enable Bi-Directional Address Translation for Your Public-Facing Servers \(Static Source NAT\)](#).

Translate Internal Client IP Addresses to Your Public IP Address (Source DIPP NAT)

When a client on your internal network sends a request, the source address in the packet contains the IP address for the client on your internal network. If you use private IP address ranges internally, the packets from the client will not be able to be routed on the Internet unless you translate the source IP address in the packets leaving the network into a publicly routable address.

On the firewall you can do this by configuring a source NAT policy that translates the source address (and optionally the port) into a public address. One way to do this is to translate the source address for all packets to the egress interface on your firewall, as shown in the following procedure.

Beginning with PAN-OS 10.2.4, you can enable [persistent NAT for DIPP](#) to mitigate the compatibility issues that symmetric NAT may have with applications that use STUN.

STEP 1 | Create an address object for the external IP address you plan to use.

1. Select **Objects > Addresses** and Add a **Name** and optional **Description** for the object.
2. Select **IP Netmask** from the **Type** and then enter the IP address of the external interface on the firewall, 203.0.113.100 in this example.
3. Click **OK**.



Although you do not have to use address objects in your policies, it is a best practice because it simplifies administration by allowing you to make updates in one place rather than having to update every policy where the address is referenced.

STEP 2 | Create the NAT policy.

1. Select **Policies > NAT** and click **Add**.
2. On the **General** tab, enter a descriptive **Name** for the policy.
3. (**Optional**) Enter a tag, which is a keyword or phrase that allows you to sort or filter policies.
4. For **NAT Type**, select **ipv4** (default).
5. On the **Original Packet** tab, select the zone you created for your internal network in the **Source Zone** section (click **Add** and then select the zone) and the zone you created for the external network from the **Destination Zone** list.
6. On the **Translated Packet** tab, select **Dynamic IP And Port** from the **Translation Type** list in the Source Address Translation section of the screen.
7. For **Address Type**, there are two choices. You could select **Translated Address** and then click **Add**. Select the address object you just created.

An alternative **Address Type** is **Interface Address**, in which case the translated address will be the IP address of the interface. For this choice, you would select an **Interface** and optionally an **IP Address** if the interface has more than one IP address.

8. Click **OK**.

STEP 3 | Commit your changes.

Click **Commit**.

STEP 4 | (**PAN-OS 10.2.4 and later 10.2 releases**) Enable persistent NAT for DIPP.

1. [Access the CLI](#).
2. > **set system setting persistent-dipp enable yes**
3. > **request restart system**
4. If you have HA configured, repeat this step on the other HA peer.

STEP 5 | (**Optional**) Access the CLI to verify the translation.

1. Use the **show session all** command to view the session table, where you can verify the source IP address and port and the corresponding translated IP address and port.
2. Use the **show session id <id_number>** to view more details about a session.
3. If you configured Dynamic IP NAT, use the **show counter global filter aspect session severity drop | match nat** command to see if any sessions failed due to NAT IP allocation. If all of the addresses in the Dynamic IP NAT pool are allocated when a new connection is supposed to be translated, the packet will be dropped.

Enable Clients on the Internal Network to Access your Public Servers (Destination U-Turn NAT)

When a user on the internal network sends a request for access to the corporate web server in the DMZ, the DNS server will resolve it to the public IP address. When processing the request, the firewall will use the original destination in the packet (the public IP address) and route the packet to the egress interface for the untrust zone. In order for the firewall to know that it must translate the public IP address of the web server to an address on the DMZ network when it

receives requests from users on the trust zone, you must create a destination NAT rule that will enable the firewall to send the request to the egress interface for the DMZ zone as follows.

STEP 1 | Create an address object for the web server.

1. Select **Objects > Addresses** and Add a Name and optional Description for the address object.
2. For Type, select **IP Netmask** and enter the public IP address of the web server, 203.0.113.11 in this example.

You can switch the address object type from **IP Netmask** to **FQDN** by clicking **Resolve**, and when the FQDN appears, click **Use this FQDN**. Alternatively, for **Type**, select **FQDN** and enter the FQDN to use for the address object. If you enter an FQDN and click **Resolve**, the IP address to which the FQDN resolves appears in the field. To switch the address object **Type** from an FQDN to an IP Netmask using this IP address, click **Use this address** and the **Type** will switch to **IP Netmask** with the IP address appearing in the field.

3. Click **OK**.

STEP 2 | Create the NAT policy.

1. Select **Policies > NAT** and click **Add**.
2. On the **General** tab, enter a descriptive **Name** for the NAT rule.
3. On the **Original Packet** tab, select the zone you created for your internal network in the **Source Zone** section (click **Add** and then select the zone) and the zone you created for the external network from the **Destination Zone** list.
4. In the **Destination Address** section, **Add** the address object you created for your public web server.
5. On the **Translated Packet** tab, for Destination Address Translation, for **Translation Type**, select **Static IP** and then enter the IP address that is assigned to the web server interface on the DMZ network, 10.1.1.11 in this example. Alternatively, you can select **Translation Type** to be **Dynamic IP (with session distribution)** and enter the **Translated Address** to be an address object or address group that uses an IP netmask, IP range, or FQDN. Any of these can return multiple addresses from DNS. If the translated destination address resolves to more than one address, the firewall distributes incoming NAT sessions among the multiple addresses based on one of several methods you can select: **Round Robin** (the default method), **Source IP Hash**, **IP Modulo**, **IP Hash**, or **Least Sessions**.
6. Click **OK**.

STEP 3 | Click **Commit**.

Enable Bi-Directional Address Translation for Your Public-Facing Servers (Static Source NAT)

When your public-facing servers have private IP addresses assigned on the network segment where they are physically located, you need a source NAT rule to translate the source address of the server to the external address upon egress. You create a static NAT rule to translate the internal source address, 10.1.1.11, to the external web server address, 203.0.113.11 in our example.

However, a public-facing server must be able to both send and receive packets. You need a reciprocal policy that translates the public address (the destination IP address in incoming packets from Internet users) into the private address so that the firewall can route the packet to your DMZ network. You create a bi-directional static NAT rule, as described in the following procedure. Bi-directional translation is an option for static NAT only.

STEP 1 | Create an address object for the web server's internal IP address.

1. Select **Objects > Addresses** and Add a **Name** and optional **Description** for the object.
2. Select **IP Netmask** from the **Type** list and enter the IP address of the web server on the DMZ network, 10.1.1.11 in this example.
3. Click **OK**.



If you did not already create an address object for the public address of your web server, you should create that object now.

STEP 2 | Create the NAT policy.

1. Select **Policies > NAT** and click **Add**.
2. On the **General** tab, enter a descriptive **Name** for the NAT rule.
3. On the **Original Packet** tab, select the zone you created for your DMZ in the **Source Zone** section (click **Add** and then select the zone) and the zone you created for the external network from the **Destination Zone** list.
4. In the **Source Address** section, **Add** the address object you created for your internal web server address.
5. On the **Translated Packet** tab, select **Static IP** from the **Translation Type** list in the **Source Address Translation** section and then select the address object you created for your external web server address from the **Translated Address** list.
6. In the **Bi-directional** field, select **Yes**.
7. Click **OK**.

STEP 3 | Commit.

Click **Commit**.

Configure Destination NAT with DNS Rewrite

When you configure a destination NAT policy rule that performs static translation of IPv4 addresses, you can also configure the rule so that the firewall rewrites the IPv4 address in a DNS response based on the original or translated IP address configured for the rule. The firewall performs NAT on the IPv4 address (the FQDN resolution) in a DNS response (that matches the rule) before forwarding the response to the client; thus, the client receives the appropriate address to reach the destination service.

View the [DNS rewrite use cases](#) to help you determine whether to specify that the rewrite occur in the **reverse** or **forward** direction.



*You cannot enable **Bi-directional** source address translation in the same NAT rule where you enable DNS rewrite.*

STEP 1 | Create a destination NAT policy rule that specifies the firewall perform static translation of IPv4 addresses that match the rule, and also specifies the firewall rewrite IP addresses in DNS responses when that IPv4 address (from the A Record) matches the original or translated destination address in the NAT rule.

1. Select **Policies > NAT** and **Add a NAT policy rule**.
2. (**Optional**) On the **General** tab, enter a descriptive **Name** for the rule.
3. For **NAT Type**, select **ipv4**.
4. On the **Original Packet** tab, **Add a Destination Address**.



You will also have to select a **Source Zone** or **Any source zone**, but DNS rewrite occurs at the global level; only the **Destination Address** on the **Original Packet tab** is matched. DNS Rewrite ignores all other fields on the **Original Packet tab**.

5. On the **Translated Packet** tab, for Destination Address Translation, select **Translation Type** to be **Static IP**.
6. Select a **Translated Address** or enter a new address.
7. **Enable DNS Rewrite** and select a **Direction**:
 - Select **reverse** (default) when the IP address in the DNS response requires the opposite translation that the NAT rule specifies. If the DNS response matches the **Translated Destination Address** in the rule, translate the DNS response using the reverse translation that the rule uses. For example, if the rule translates IP address 1.1.1.10 to 192.168.1.10, the firewall rewrites a DNS response of 192.168.1.10 to 1.1.1.10.
 - Select **forward** when the IP address in the DNS response requires the same translation that the NAT rule specifies. If the DNS response matches the **Original Destination Address** in the rule, translate the DNS response using the same translation the rule uses. For example, if the rule translates IP address 1.1.1.10 to 192.168.1.10, the firewall rewrites a DNS response of 1.1.1.10 to 192.168.1.10.
8. Click **OK**.

STEP 2 | Commit your changes.

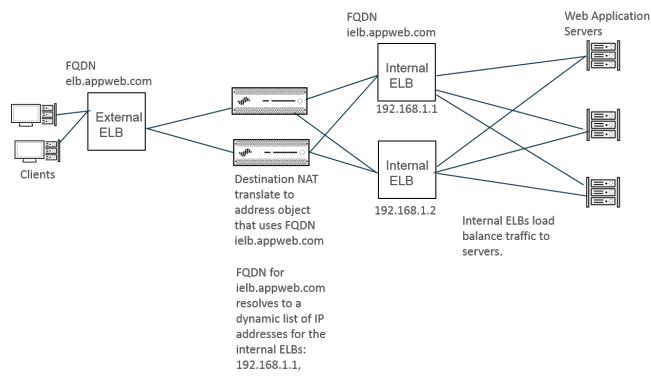
Configure Destination NAT Using Dynamic IP Addresses

Use **Destination NAT** to translate the original destination address to a destination host or server that has a dynamic IP address and uses an FQDN. Destination NAT using a dynamic IP address is especially helpful in cloud deployments, which typically use dynamic IP addressing. When the host or server in the cloud has new (dynamic) IP addresses, you don't need to manually update the NAT policy rule by continuously querying the DNS server, nor do you need to use a separate, external component to update the DNS server with the latest FQDN-to-IP address mapping.

When you configure destination NAT using dynamic IP addresses, you should use only an FQDN (not an IP netmask or IP range).

In the following example topology, clients want to reach servers that are hosting web applications in the cloud. An external Elastic Load Balancer (ELB) connects to firewalls, which connect to internal ELBs that connect to the servers. Over time, Amazon Web Services (AWS), for example, adds (and removes) IP addresses for the FQDN assigned to the internal ELBs based on the demand for services. The flexibility of using an FQDN for NAT to the internal ELB helps the policy

to resolve to different IP addresses at different times, making destination NAT easier to use because the updates are dynamic.



STEP 1 | Create an address object using the FQDN of the server to which you want to translate the address.

1. Select **Objects > Addresses** and Add an address object by **Name**, such as **post-NAT-Internal-ELB**.
2. Select **FQDN** as the **Type** and enter the FQDN. In this example, the FQDN is **ielb.appweb.com**.
3. Click **OK**.

STEP 2 | Create the destination NAT policy.

1. Select **Policies > NAT** and Add a NAT policy rule by **Name** on the **General** tab.
2. Select **ipv4** as the **NAT Type**.
3. On the **Original Packet** tab, Add the **Source Zone** and **Destination Zone**.
4. On the **Translated Packet** tab, in the **Destination Address Translation** section, select **Dynamic IP (with session distribution)** as the **Translation Type**.
5. For **Translated Address**, select the address object you created for the FQDN. In this example, the FQDN is **post-NAT-Internal-ELB**.
6. For **Session Distribution Method**, select one of the following:
 - **Round Robin** (default)—Assigns new sessions to IP addresses in rotating order. Unless you have a reason to change the distribution method, round robin distribution is likely suitable.
 - **Source IP Hash**—Assigns new sessions based on hash of source IP address. If you have traffic coming from a single source IP address, don't select Source IP Hash; select a different method.
 - **IP Modulo**—The firewall takes into consideration the source and destination IP address from the incoming packet; the firewall performs an XOR operation and a

modulo operation; the result determines to which IP address the firewall assigns new sessions.

- **IP Hash**—Assigns new sessions based on hash of source and destination IP addresses.
- **Least Sessions**—Assigns new sessions to the IP address with the fewest concurrent sessions. If you have many short-lived sessions, **Least Sessions** provides you with a more balanced distribution of sessions.



The firewall does not remove duplicate IP addresses from the list of destination IP addresses before it distributes sessions among the multiple IP addresses.

The firewall distributes sessions to the duplicate addresses in the same way it distributes sessions to non-duplicate addresses. (Duplicate addresses in the translation pool can occur, for example, if the translated address is an address group of address objects, and one address object is an FQDN that resolves to an IP address, while another address object is a range that includes the same IP address.)

7. Click **OK**.

STEP 3 | Commit your changes.

STEP 4 | (Optional) You can configure the frequency at which the firewall refreshes an FQDN ([Use Case 1: Firewall Requires DNS Resolution](#)).

Modify the Oversubscription Rate for DIPP NAT

If you have enough public IP addresses that you do not need to use DIPP NAT oversubscription, you can reduce the oversubscription rate and thereby gain more DIP and DIPP NAT rules allowed.

STEP 1 | View the DIPP NAT oversubscription rate.

1. Select **Device > Setup > Session > Session Settings**. View the **NAT Oversubscription Rate** setting.

STEP 2 | Set the DIPP NAT oversubscription rate.

1. Edit the Session Settings section.
2. In the **NAT Oversubscription Rate** list, select **1x**, **2x**, **4x**, or **8x**, depending on which ratio you want.



*The Platform Default setting applies the default oversubscription setting for the model. If you want no oversubscription, select **1x**.*

3. Click **OK** and **Commit** the change.

Reserve Dynamic IP NAT Addresses

You can reserve Dynamic IP NAT addresses (for a configurable period of time) to prevent them from being allocated as translated addresses to a different source IP address that needs translation. When configured, the reservation applies to all of the translated Dynamic IP addresses in progress and any new translations.

For both translations in progress and new translations, when a source IP address is translated to an available translated IP address, that pairing is retained even after all sessions related to

that specific source IP are expired. The reservation timer for each source IP address begins after all sessions that use that source IP address translation expire. Dynamic IP NAT is a one-to-one translation; one source IP address translates to one translated IP address that is chosen dynamically from those addresses available in the configured pool. Therefore, a translated IP address that is reserved is not available for any other source IP address until the reservation expires because a new session has not started. The timer is reset each time a new session for a source IP/translated IP mapping begins, after a period when no sessions were active.

By default, no addresses are reserved. You can reserve Dynamic IP NAT addresses for the firewall or for a virtual system.

- Reserve dynamic IP NAT addresses for a firewall.

Enter the following commands:

```
admin@PA-3250# set setting nat reserve-ip yes
```

```
admin@PA-3250# set setting nat reserve-time <1-604800 secs>
```

- Reserve dynamic IP NAT addresses for a virtual system.

Enter the following commands:

```
admin@PA-3250# set vsys <vsydid> setting nat reserve-ip yes
```

```
admin@PA-3250# set vsys <vsydid> setting nat reserve-time <1-604800 secs>
```

For example, suppose there is a Dynamic IP NAT pool of 30 addresses and there are 20 translations in progress when the **nat reserve-time** is set to 28800 seconds (8 hours). Those 20 translations are now reserved, so that when the last session (of any application) that uses each source IP/translated IP mapping expires, the translated IP address is reserved for only that source IP address for 8 hours, in case that source IP address needs translation again. Additionally, as the 10 remaining translated addresses are allocated, they each are reserved for their source IP address, each with a timer that begins when the last session for that source IP address expires.

In this manner, each source IP address can be repeatedly translated to its same NAT address from the pool; another host will not be assigned a reserved translated IP address from the pool, even if there are no active sessions for that translated address.

Suppose a source IP/translated IP mapping has all of its sessions expire, and the reservation timer of 8 hours begins. After a new session for that translation begins, the timer stops, and

the sessions continue until they all end, at which point the reservation timer starts again, reserving the translated address.

The reservation timer remain in effect on the Dynamic IP NAT pool until you disable it by entering the **set setting nat reserve-ip no** command or you change the **nat reserve-time** to a different value.

The CLI commands for reservations do not affect Dynamic IP and Port (DIPP) or Static IP NAT pools.

Disable NAT for a Specific Host or Interface

Both source NAT and destination NAT rules can be configured to disable address translation. You may have exceptions where you do not want NAT to occur for a certain host in a subnet or for traffic exiting a specific interface. The following procedure shows how to disable source NAT for a host.

STEP 1 | Create the NAT policy.

1. Select **Policies > NAT** and click **Add** a descriptive **Name** for the policy.
2. On the **Original Packet** tab, select the zone you created for your internal network in the **Source Zone** section (click **Add** and then select the zone) and the zone you created for the external network from the **Destination Zone** list.
3. For **Source Address**, click **Add** and enter the host address. Click **OK**.
4. On the **Translated Packet** tab, select **None** from the **Translation Type** list in the Source Address Translation section of the screen.
5. Click **OK**.

STEP 2 | Commit your changes.

Click **Commit**.



NAT rules are processed in order from the top to the bottom, so place the NAT exemption policy before other NAT policies to ensure it is processed before an address translation occurs for the sources you want to exempt.

NAT Configuration Examples

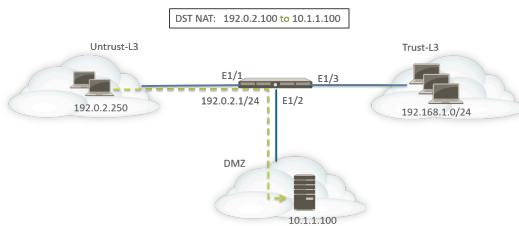
- [Destination NAT Example—One-to-One Mapping](#)
- [Destination NAT with Port Translation Example](#)
- [Destination NAT Example—One-to-Many Mapping](#)
- [Source and Destination NAT Example](#)
- [Virtual Wire Source NAT Example](#)
- [Virtual Wire Static NAT Example](#)
- [Virtual Wire Destination NAT Example](#)

Destination NAT Example—One-to-One Mapping

The most common mistakes when configuring NAT and security rules are the references to the zones and address objects. The addresses used in destination NAT rules always refer to the original IP address in the packet (that is, the pre-translated address). The destination zone in the NAT rule is determined after the route lookup of the destination IP address in the original packet (that is, the pre-NAT destination IP address).

The addresses in the security policy also refer to the IP address in the original packet (that is, the pre-NAT address). However, the destination zone is the zone where the end host is physically connected. In other words, the destination zone in the security rule is determined after the route lookup of the post-NAT destination IP address.

In the following example of a one-to-one destination NAT mapping, users from the zone named Untrust-L3 access the server 10.1.1.100 in the zone named DMZ using the IP address 192.0.2.100.



Before configuring the NAT rules, consider the sequence of events for this scenario.

- ❑ Host 192.0.2.250 sends an ARP request for the address 192.0.2.100 (the public address of the destination server).
- ❑ The firewall receives the ARP request packet for destination 192.0.2.100 on the Ethernet1/1 interface and processes the request. The firewall responds to the ARP request with its own MAC address because of the destination NAT rule configured.
- ❑ The NAT rules are evaluated for a match. For the destination IP address to be translated, a destination NAT rule from zone Untrust-L3 to zone Untrust-L3 must be created to translate the destination IP of 192.0.2.100 to 10.1.1.100.
- ❑ After determining the translated address, the firewall performs a route lookup for destination 10.1.1.100 to determine the egress interface. In this example, the egress interface is Ethernet1/2 in zone DMZ.

- The firewall performs a security policy lookup to see if the traffic is permitted from zone Untrust-L3 to DMZ.

 *The direction of the policy matches the ingress zone and the zone where the server is physically located.*

 *The security policy refers to the IP address in the original packet, which has a destination address of 192.0.2.100.*

- The firewall forwards the packet to the server out egress interface Ethernet1/2. The destination address is changed to 10.1.1.100 as the packet leaves the firewall.

For this example, address objects are configured for webserver-private (10.1.1.100) and Webserver-public (192.0.2.100). The configured NAT rule would look like this:

NAME	TAGS	Original Packet						Translated Packet	
		SOURCE ZONE	DESTINATION ZONE	DESTINATION INTERFACE	SOURCE ADDRESS	DESTINATION ADDRESS	SERVICE	SOURCE TRANSLATION	DESTINATION TRANSLATION
Dist NAT-webserver	none			any	any			any	none destination-translation address: webserver-private

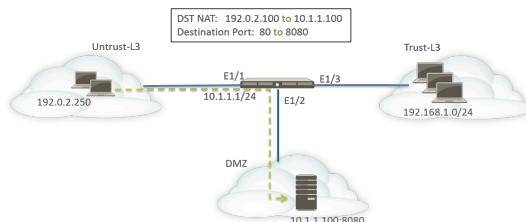
The direction of the NAT rules is based on the result of route lookup.

The configured security policy to provide access to the server from the Untrust-L3 zone would look like this:

NAME	Source		Destination		APPLICATION	SERVICE	ACTION	PROFILE	OPTIONS
	ZONE	ADDRESS	ZONE	ADDRESS					
Webserver access		any			web-browsing	any		none	

Destination NAT with Port Translation Example

In this example, the web server is configured to listen for HTTP traffic on port 8080. The clients access the web server using the IP address 192.0.2.100 and TCP Port 80. The destination NAT rule is configured to translate both IP address and port to 10.1.1.100 and TCP port 8080. Address objects are configured for webserver-private (10.1.1.100) and Servers-public (192.0.2.100).



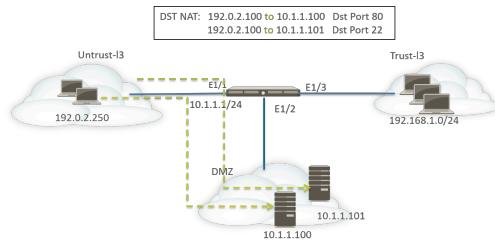
The following NAT and security rules must be configured on the firewall:

NAME	TAGS	Original Packet						Translated Packet				
		SOURCE ZONE	DESTINATION ZONE	DESTINATION INTERFACE	SOURCE ADDRESS	DESTINATION ADDRESS	SERVICE	SOURCE TRANSLATION	DESTINATION TRANSLATION			
Dist NAT-webserver	none			any	any			any	none destination-translation address: webserver-private port: 8080			
<hr/>												
NAME	TAGS	TYPE	ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE	APPLICATION	SERVICE	ACTION
Webserver access	none	universal		any	any	any			any		any	

Use the **show session all** CLI command to verify the translation.

Destination NAT Example—One-to-Many Mapping

In this example, one IP address maps to two different internal hosts. The firewall uses the application to identify the internal host to which the firewall forwards the traffic.



All HTTP traffic is sent to host 10.1.1.100 and SSH traffic is sent to server 10.1.1.101. The following address objects are required:

- Address object for the one pre-translated IP address of the server
- Address object for the real IP address of the SSH server
- Address object for the real IP address of the web server

The corresponding address objects are created:

- Servers-public: 192.0.2.100
- SSH-server: 10.1.1.101
- webserver-private: 10.1.1.100

The NAT rules would look like this:

NAME	TAGS	Original Packet						Translated Packet	
		SOURCE ZONE	DESTINATION ZONE	DESTINATION INTERFACE	SOURCE ADDRESS	DESTINATION ADDRESS	SERVICE	SOURCE TRANSLATION	DESTINATION TRANSLATION
Dst NAT-webserver	none	Untrust-L3	Untrust-L3	any	any	Servers-public	service-http	none	destination-translation address: webserver-private
Dst NAT-SSH	none	Untrust-L3	Untrust-L3	any	any	Servers-public	custom-ssh	none	destination-translation address: SSH-server

The security rules would look like this:

NAME	TAGS	TYPE	Source				Destination				APPLICATION	SERVICE	ACTION
			ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE				
Webserver access	none	universal	Untrust-L3	any	any	any	DMZ	Servers-public	any	web-browsing	any	Allow	
SSH access	none	universal	Untrust-L3	any	any	any	DMZ	Servers-public	any	ssh	any	Allow	

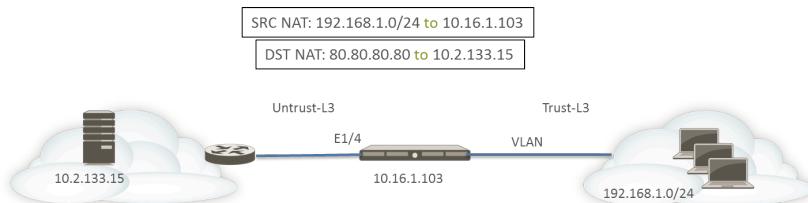
Source and Destination NAT Example

In this example, NAT rules translate both the source and destination IP address of packets between the clients and the server.

- Source NAT—The source addresses in the packets from the clients in the Trust-L3 zone to the server in the Untrust-L3 zone are translated from the private addresses in the network 192.168.1.0/24 to the IP address of the egress interface on the firewall (10.16.1.103). Dynamic IP and Port translation causes the port numbers to be translated also.

NAT

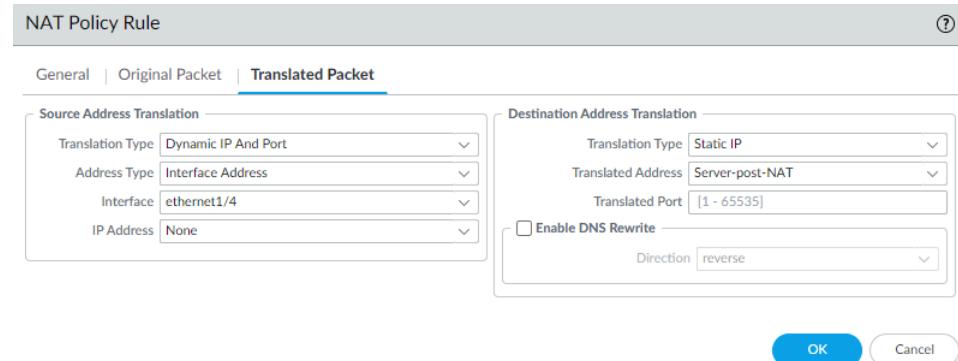
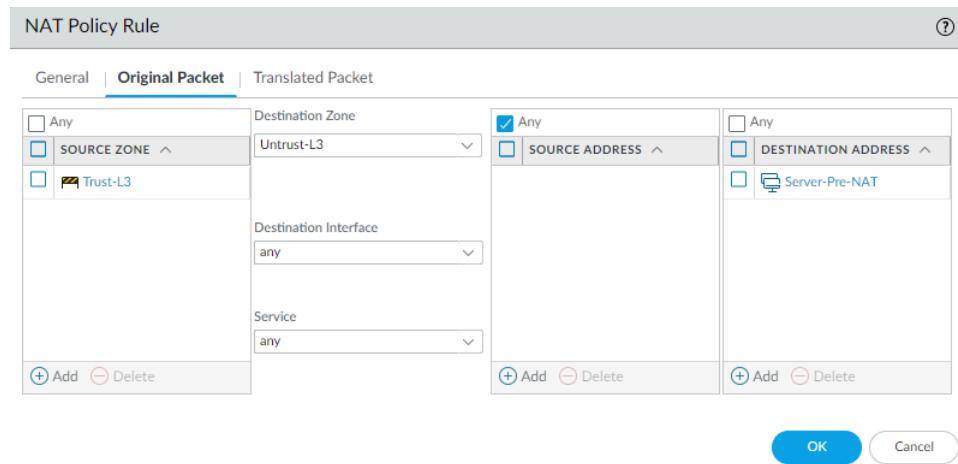
- Destination NAT—The destination addresses in the packets from the clients to the server are translated from the server's public address (80.80.80.80) to the server's private address (10.2.133.15).



The following address objects are created for destination NAT.

- Server-Pre-NAT: 80.80.80.80
- Server-post-NAT: 10.2.133.15

The following screen shots illustrate how to configure the source and destination NAT policies for the example.



To verify the translations, use the CLI command **show session all filter destination 80.80.80.80**. A client address 192.168.1.11 and its port number are translated to 10.16.1.103 and a port number. The destination address 80.80.80.80 is translated to 10.2.133.15.

Virtual Wire Source NAT Example

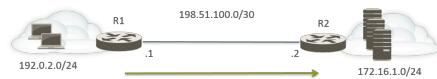
Virtual wire deployment of a Palo Alto Networks® firewall includes the benefit of providing security transparently to the end devices. It is possible to configure NAT for interfaces configured in a virtual wire. All of the NAT types are allowed: source NAT (Dynamic IP, Dynamic IP and Port, static) and destination NAT.

Because interfaces in a virtual wire do not have an IP address assigned, it is not possible to translate an IP address to an interface IP address. You must configure an IP address pool.

When performing NAT on virtual wire interfaces, it is recommended that you translate the source address to a different subnet than the one on which the neighboring devices are communicating. The firewall will not proxy ARP for NAT addresses. Proper routing must be configured on the upstream and downstream routers in order for the packets to be translated in virtual wire mode. Neighboring devices will only be able to resolve ARP requests for IP addresses that reside on the interface of the device on the other end of the virtual wire. See [Proxy ARP for NAT Address Pools](#) for more explanation about proxy ARP.

In the source NAT example below, security policies (not shown) are configured from the virtual wire zone named vw-trust to the zone named vw-untrust.

In the following topology, two routers are configured to provide connectivity between subnets 192.0.2.0/24 and 172.16.1.0/24. The link between the routers is configured in subnet 198.51.100.0/30. Static routing is configured on both routers to establish connectivity between the networks. Before the firewall is deployed in the environment, the topology and the routing table for each router look like this:



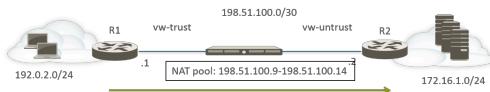
Route on R1:

Destination	Next Hop
172.16.1.0/24	198.51.100.2

Route on R2:

Destination	Next Hop
192.0.2.0/24	198.51.100.1

Now the firewall is deployed in virtual wire mode between the two Layer 3 devices. A NAT IP address pool with range 198.51.100.9 to 198.51.100.14 is configured on the firewall. All communications from clients in subnet 192.0.2.0/24 accessing servers in network 172.16.1.0/24 will arrive at R2 with a translated source address in the range 198.51.100.9 to 198.51.100.14. The response from servers will be directed to these addresses.



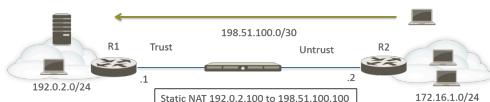
In order for source NAT to work, you must configure proper routing on R2, so that packets destined for other addresses are not dropped. The routing table below shows the modified routing table on R2; the route ensures traffic to the destinations 198.51.100.9-198.51.100.14 (that is, hosts on subnet 198.51.100.8/29) will be sent back through the firewall to R1.

Route on R2:

Destination	Next Hop
198.51.100.8/29	198.51.100.1

Virtual Wire Static NAT Example

In this example, security policies are configured from the virtual wire zone named Trust to the virtual wire zone named Untrust. Host 192.0.2.100 is statically translated to address 198.51.100.100. With the **Bi-directional** option enabled, the firewall generates a NAT policy from the Untrust zone to the Trust zone. Clients on the Untrust zone access the server using the IP address 198.51.100.100, which the firewall translates to 192.0.2.100. Any connections initiated by the server at 192.0.2.100 are translated to source IP address 198.51.100.100.



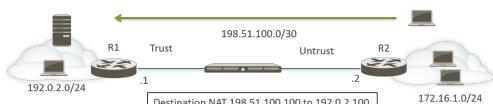
Route on R2:

Destination	Next Hop
198.51.100.100/32	198.51.100.1

NAME	Original Packet						Translated Packet	
	SOURCE ZONE	DESTINATION ZONE	DESTINATION INTERFACE	SOURCE ADDRESS	DESTINATION ADDRESS	SERVICE	SOURCE TRANSLATION	DESTINATION TRANSLATION
Static NAT	trust	Untrust	any	webservice-private	any	any	static-ip webservice-public bi-directional: yes	none

Virtual Wire Destination NAT Example

Clients in the Untrust zone access the server using the IP address 198.51.100.100, which the firewall translates to 192.0.2.100. Both the NAT and security policies must be configured from the Untrust zone to the Trust zone.



Route on R2:

Destination	Next Hop
198.51.100.100/32	198.51.100.1

NAME	Original Packet						Translated Packet	
	SOURCE ZONE	DESTINATION ZONE	DESTINATION INTERFACE	SOURCE ADDRESS	DESTINATION ADDRESS	SERVICE	SOURCE TRANSLATION	DESTINATION TRANSLATION
DST NAT	Untrust	Trust	any	any	webserver-public	any	none	destination-translation address: webserver-private

NPTv6

IPv6-to-IPv6 Network Prefix Translation (NPTv6) performs a stateless, static translation of one IPv6 prefix to another IPv6 prefix (port numbers are not changed). There are four primary benefits of NPTv6:

- You can prevent the asymmetrical routing problems that result from Provider Independent addresses being advertised from multiple datacenters.
- NPTv6 allows more specific routes to be advertised so that return traffic arrives at the same firewall that transmitted the traffic.
- Private and public addresses are independent; you can change one without affecting the other.
- You have the ability to translate [Unique Local Addresses](#) to globally routable addresses.

This topic builds on a basic understanding of NAT. You should be sure you are familiar with [NAT](#) concepts before configuring NPTv6.

- [NPTv6 Overview](#)
- [How NPTv6 Works](#)
- [NDP Proxy](#)
- [NPTv6 and NDP Proxy Example](#)
- [Create an NPTv6 Policy](#)

NPTv6 Overview

This section describes [IPv6-to-IPv6 Network Prefix Translation](#) (NPTv6) and how to configure it. NPTv6 is defined in [RFC 6296](#). Palo Alto Networks® does not implement all functionality defined in the RFC, but is compliant with the RFC in the functionality it has implemented.

NPTv6 performs stateless translation of one IPv6 prefix to another IPv6 prefix. It is stateless, meaning that it does not keep track of ports or sessions on the addresses translated. NPTv6 differs from NAT66, which is stateful. Palo Alto Networks supports [NPTv6 RFC 6296](#) prefix translation; it does not support NAT66.

With the limited addresses in the IPv4 space, [NAT](#) was required to translate private, non-routable IPv4 addresses to one or more globally-routable IPv4 addresses. For organizations using IPv6 addressing, there is no need to translate IPv6 addresses to IPv6 addresses due to the abundance of IPv6 addresses. However, there are [Reasons to Use NPTv6](#) to translate IPv6 prefixes at the firewall.



It is important to understand that NPTv6 does not provide security. In general, stateless network address translation does not provide any security; it provides an address translation function. NPTv6 does not hide or translate port numbers. You must set up firewall security policies correctly in each direction to ensure that traffic is controlled as you intended.

NPTv6 translates the prefix portion of an IPv6 address but not the host portion or the application port numbers. The host portion is simply copied, and therefore remains the same on either side of the firewall. The host portion also remains visible within the packet header.

NPTv6 is supported on the following firewall models (NPTv6 with hardware lookup but packets go through the CPU):

- PA-7000 Series firewalls
- PA-5400 Series firewalls
- PA-5200 Series firewalls
- PA-3200 Series firewalls
- PA-800 firewall
- PA-220 firewall

VM-Series firewalls support NPTv6, but with no ability to have hardware perform a session lookup.

- [Unique Local Addresses](#)
- [Reasons to Use NPTv6](#)

Unique Local Addresses

[RFC 4193, Unique Local IPv6 Unicast Addresses](#), defines unique local addresses (ULAs), which are IPv6 unicast addresses. They can be considered IPv6 equivalents of the private IPv4 addresses identified in [RFC 1918, Address Allocation for Private Internets](#), which cannot be routed globally.

A ULA is globally unique, but not expected to be globally routable. It is intended for local communications and to be routable in a limited area such as a site or among a small number of sites. Palo Alto Networks® does not recommend that you assign ULAs, but a firewall configured with NPTv6 will translate prefixes sent to it, including ULAs.

Reasons to Use NPTv6

Although there is no shortage of public, globally routable IPv6 addresses, there are reasons you might want to translate IPv6 addresses. NPTv6:

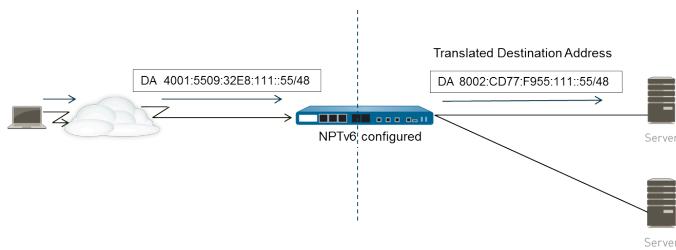
- **Prevents asymmetrical routing**—Asymmetric routing can occur if a Provider Independent address space (/48, for example) is advertised by multiple data centers to the global Internet. By using NPTv6, you can advertise more specific routes from regional firewalls, and the return traffic will arrive at the same firewall where the source IP address was translated by the translator.
- **Provides address independence**—You need not change the IPv6 prefixes used inside your local network if the global prefixes are changed (for example, by an ISP or as a result of merging organizations). Conversely, you can change the inside addresses at will without disrupting the addresses that are used to access services in the private network from the Internet. In either case, you update a NAT rule rather than reassign network addresses.
- **Translates ULAs for routing**—You can have [Unique Local Addresses](#) assigned within your private network, and have the firewall translate them to globally routable addresses. Thus, you have the convenience of private addressing and the functionality of translated, routable addresses.
- **Reduces exposure to IPv6 prefixes**—IPv6 prefixes are less exposed than if you didn't translate network prefixes, however, NPTv6 is not a security measure. The interface identifier portion of each IPv6 address is not translated; it remains the same on each side of the firewall and visible to anyone who can see the packet header. Additionally, the prefixes are not secure; they can be determined by others.

How NPTv6 Works

When you configure a policy for NPTv6, the Palo Alto Networks® firewall performs a static, one-to-one IPv6 translation in both directions. The translation is based on the algorithm described in [RFC 6296](#).

In one use case, the firewall performing NPTv6 is located between an internal network and an external network (such as the Internet) that uses globally routable prefixes. When datagrams are going in the outbound direction, the internal source prefix is replaced with the external prefix; this is known as source translation.

In another use case, when datagrams are going in the inbound direction, the destination prefix is replaced with the internal prefix (known as destination translation). The figure below illustrates destination translation and a characteristic of NPTv6: only the prefix portion of an IPv6 address is translated. The host portion of the address is not translated and remains the same on either side of the firewall. In the figure below, the host identifier is 111::55 on both sides of the firewall.



It is important to understand that NPTv6 does not provide security. While you are planning your NPTv6 NAT policies, remember also to configure security policies in each direction.

A NAT or NPTv6 policy rule cannot have both the Source Address and the Translated Address set to Any.

In an environment where you want IPv6 prefix translation, three firewall features work together: NPTv6 NAT policies, security policies, and [NDP Proxy](#).

The firewall does not translate the following:

- Addresses that the firewall has in its Neighbor Discovery (ND) cache.
- The subnet 0xFFFF (in accordance with [RFC 6296](#), Appendix B).
- IP multicast addresses.
- IPv6 addresses with a prefix length of /31 or shorter.
- Link-local addresses. If the firewall is operating in virtual wire mode, there are no IP addresses to translate, and the firewall does not translate link-local addresses.
- Addresses for TCP sessions that authenticate peers using the TCP Authentication Option ([RFC 5925](#)).

When using NPTv6, performance for fast path traffic is impacted because NPTv6 is performed in the slow path.

NPTv6 will work with IPSec IPv6 only if the firewall is originating and terminating the tunnel. Transit IPSec traffic would fail because the source and/or destination IPv6 address would be modified. A NAT traversal technique that encapsulates the packet would allow IPSec IPv6 to work with NPTv6.

- Checksum-Neutral Mapping
- Bi-Directional Translation
- NPTv6 Applied to a Specific Service

Checksum-Neutral Mapping

The NPTv6 mapping translations that the firewall performs are checksum-neutral, meaning that “... they result in IP headers that will generate the same IPv6 pseudo-header checksum when the checksum is calculated using the standard Internet checksum algorithm [RFC 1071].” See [RFC 6296](#), Section 2.6, for more information about checksum-neutral mapping.

If you are using NPTv6 to perform destination NAT, you can provide the internal IPv6 address and the external prefix/prefix length of the firewall interface in the syntax of the `test nptv6` CLI command. The CLI responds with the checksum-neutral, public IPv6 address to use in your NPTv6 configuration to reach that destination.

Bi-Directional Translation

When you [Create an NPTv6 Policy](#), the **Bi-directional** option in the **Translated Packet** tab provides a convenient way for you to have the firewall create a corresponding NAT or NPTv6 translation in the opposite direction of the translation you configured. By default, **Bi-directional** translation is disabled.



*If you enable **Bi-directional** translation, it is very important to make sure you have security policies in place to control the traffic in both directions. Without such policies, the **Bi-directional** feature will allow packets to be automatically translated in both directions, which you might not want.*

NPTv6 Applied to a Specific Service

The Palo Alto Networks implementation of NPTv6 offers the ability to filter packets to limit which packets are subject to translation. Keep in mind that NPTv6 does not perform port translation. There is no concept of Dynamic IP and Port (DIPP) translation because NPTv6 translates IPv6 prefixes only. However, you can specify that only packets for a certain service port undergo NPTv6 translation. To do so, [Create an NPTv6 Policy](#) that specifies a **Service** in the Original Packet.

NDP Proxy

Neighbor Discovery Protocol (NDP) for IPv6 performs functions similar to those provided by Address Resolution Protocol (ARP) for IPv4. [RFC 4861](#) defines [Neighbor Discovery for IP version 6 \(IPv6\)](#). Hosts, routers, and firewalls use NDP to determine the link-layer addresses of neighbors on connected links, to keep track of which neighbors are reachable, and to update neighbors' link-layer addresses that have changed. Peers advertise their own MAC address and IPv6 address, and they also solicit addresses from peers.

NDP also supports the concept of *proxy*, when a node has a neighboring device that is able to forward packets on behalf of the node. The device (firewall) performs the role of NDP Proxy.

Palo Alto Networks® firewalls support NDP and NDP Proxy on their interfaces. When you configure the firewall to act as an NDP Proxy for addresses, it allows the firewall to send Neighbor Discovery (ND) advertisements and respond to ND solicitations from peers that are asking for MAC addresses of IPv6 prefixes assigned to devices behind the firewall. You can also configure addresses for which the firewall will not respond to proxy requests (negated addresses).

In fact, NDP is enabled by default, and you need to configure NDP Proxy when you configure NPTv6, for the following reasons:

- The stateless nature of NPTv6 requires a way to instruct the firewall to respond to ND packets sent to specified NDP Proxy addresses, and to not respond to negated NDP Proxy addresses.



It is recommended that you negate your neighbors' addresses in the NDP Proxy configuration, because NDP Proxy indicates the firewall will reach those addresses behind the firewall, but the neighbors are not behind the firewall.

- NDP causes the firewall to save the MAC addresses and IPv6 addresses of neighbors in its ND cache. (Refer to the figure in [NPTv6 and NDP Proxy Example](#).) The firewall does not perform NPTv6 translation for addresses that it finds in its ND cache because doing so could introduce a conflict. If the host portion of an address in the cache happens to overlap with the host portion of a neighbor's address, and the prefix in the cache is translated to the same prefix as that of the neighbor (because the egress interface on the firewall belongs to the same subnet as the neighbor), then you would have a translated address that is exactly the same as the legitimate IPv6 address of the neighbor, and a conflict occurs. (If an attempt to perform NPTv6 translation occurs on an address in the ND cache, an informational syslog message logs the event: **NPTv6 Translation Failed**.)

When an interface with NDP Proxy enabled receives an ND solicitation requesting a MAC address for an IPv6 address, the following sequence occurs:

- ❑ The firewall searches the ND cache to ensure the IPv6 address from the solicitation is not there. If the address is there, the firewall ignores the ND solicitation.
- ❑ If the source IPv6 address is 0, that means the packet is a Duplicate Address Detection packet, and the firewall ignores the ND solicitation.
- ❑ The firewall does a Longest Prefix Match search of the NDP Proxy addresses and finds the best match to the address in the solicitation. If the Negate field for the match is checked (in the NDP Proxy list), the firewall drops the ND solicitation.
- ❑ Only if the Longest Prefix Match search matches, and that matched address is not negated, will the NDP Proxy respond to the ND solicitation. The firewall responds with an ND packet,

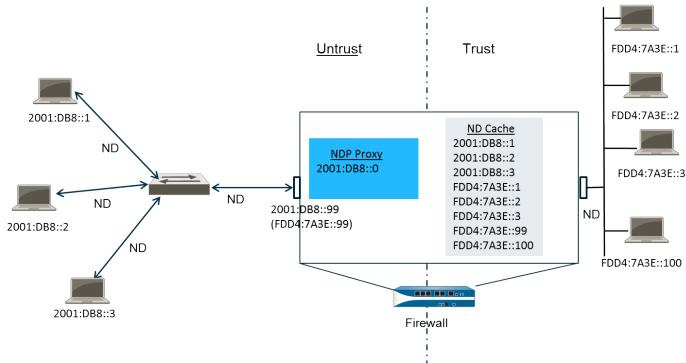
providing its own MAC address as the MAC address of the next hop toward the queried destination.

In order to successfully support NDP, the firewall does not perform NDP Proxy for the following:

- Duplicate Address Detection (DAD).
- Addresses in the ND cache (because such addresses do not belong to the firewall; they belong to discovered neighbors).

NPTv6 and NDP Proxy Example

The following figure illustrates how NPTv6 and NDP Proxy function together.



- [The ND Cache in NPTv6 Example](#)
- [The NDP Proxy in NPTv6 Example](#)
- [The NPTv6 Translation in NPTv6 Example](#)
- [Neighbors in the ND Cache are Not Translated](#)

The ND Cache in NPTv6 Example

In the above example, multiple peers connect to the firewall through a switch, with ND occurring between the peers and the switch, between the switch and the firewall, and between the firewall and the devices on the trust side.

As the firewall learns of peers, it saves their addresses to its ND cache. Trusted peers FDDA:7A3E::1, FDDA:7A3E::2, and FDDA:7A3E::3 are connected to the firewall on the trust side. FDDA:7A3E::99 is the untranslated address of the firewall itself; its public-facing address is 2001:DB8::99. The addresses of the peers on the untrust side have been discovered and appear in the ND cache: 2001:DB8::1, 2001:DB8::2, and 2001:DB8::3.

The NDP Proxy in NPTv6 Example

In our scenario, we want the firewall to act as NDP Proxy for the prefixes on devices behind the firewall. When the firewall is NDP Proxy for a specified set of addresses/ranges/prefixes, and it sees an address from this range in an ND solicitation or advertisement, the firewall will respond as long as a device with that specific address doesn't respond first, the address is not negated in the NDP proxy configuration, and the address is not in the ND cache. The firewall does the prefix translation (described below) and sends the packet to the trust side, where that address might or might not be assigned to a device.

In this example, the ND Proxy table contains the network address 2001:DB8::0. When the interface sees an ND for 2001:DB8::100, no other devices on the L2 switch claim the packet, so the proxy range causes the firewall to claim it, and after translation to FDD4:7A3E::100, the firewall sends it out to the trust side.

The NPTv6 Translation in NPTv6 Example

In this example, the **Original Packet** is configured with a **Source Address** of FDD4:7A3E::0 and a **Destination of Any**. The **Translated Packet** is configured with the **Translated Address** of 2001:DB8::0.

Therefore, outgoing packets with a source of FDD4:7A3E::0 are translated to 2001:DB8::0. Incoming packets with a destination prefix in the network 2001:DB8::0 are translated to FDD4:7A3E::0.

Neighbors in the ND Cache are Not Translated

In our example, there are hosts behind the firewall with host identifiers :1, :2, and :3. If the prefixes of those hosts are translated to a prefix that exists beyond the firewall, and if those devices also have host identifiers :1, :2, and :3, because the host identifier portion of the address remains unchanged, the resulting translated address would belong to the existing device, and an addressing conflict would result. In order to avoid a conflict with overlapping host identifiers, NPTv6 does not translate addresses that it finds in its ND cache.

Create an NPTv6 Policy

Perform this task when you want to configure a NAT [NPTv6](#) policy to translate one IPv6 prefix to another IPv6 prefix. The prerequisites for this task are:

- Enable IPv6. Select **Device > Setup > Session**. Click **Edit** and select **IPv6 Firewalling**.
- Configure a Layer 3 Ethernet interface with a valid IPv6 address and with IPv6 enabled. Select **Network > Interfaces > Ethernet**, select an interface, and on the **IPv6** tab, select **Enable IPv6 on the interface**.
- Create network security policies, because NPTv6 does not provide security.
- Decide whether you want source translation, destination translation, or both.
- Identify the zones to which you want to apply the NPTv6 policy.
- Identify your original and translated IPv6 prefixes.

STEP 1 | Create a new NPTv6 policy.

1. Select **Policies > NAT** and click **Add**.
2. On the **General** tab, enter a descriptive **Name** for the NPTv6 policy rule.
3. **(Optional)** Enter a **Description** and **Tag**.
4. For **NAT Type**, select **NPTv6**.

STEP 2 | Specify the match criteria for incoming packets; packets that match all of the criteria are subject to the NPTv6 translation.

Zones are required for both types of translation.

1. On the **Original Packet** tab, for **Source Zone**, leave **Any** or **Add** the source zone to which the policy applies.
2. Enter the **Destination Zone** to which the policy applies.
3. **(Optional)** Select a **Destination Interface**.
4. **(Optional)** Select a **Service** to restrict what type of packets are translated.
5. If you are doing source translation, enter a **Source Address** or select **Any**. The address could be an address object. The following constraints apply to **Source Address** and **Destination Address**:
 - Prefixes of **Source Address** and **Destination Address** for the **Original Packet** and **Translated Packet** must be in the format xxxx:xxxx::/yy, although leading zeros in the prefix can be dropped.
 - The IPv6 address cannot have an interface identifier (host) portion defined.
 - The range of supported prefix lengths is /32 to /112.
 - The **Source Address** and **Destination Address** cannot both be set to **Any**.
6. If you are doing source translation, you can optionally enter a **Destination Address**. If you are doing destination translation, the **Destination Address** is required. The destination address (an address object is allowed) must be a netmask, not just an IPv6 address and not a range. The prefix length must be a value from /32 to /112, inclusive. For example, 2001:db8::/32.

STEP 3 | Specify the translated packet.

1. On the **Translated Packet** tab, if you want to do source translation, in the Source Address Translation section, for **Translation Type**, select **Static IP**. If you do not want to do source translation, select **None**.
2. If you chose **Static IP**, the **Translated Address** field appears. Enter the translated IPv6 prefix or address object. See the constraints listed in the prior step.



*It is a best practice to configure your **Translated Address** to be the prefix of the untrust interface address of your firewall. For example, if your untrust interface has the address 2001:1a:1b:1::99/64, make your **Translated Address** 2001:1a:1b:1::0/64.*

3. **(Optional)** Select **Bi-directional** if you want the firewall to create a corresponding NPTv6 translation in the opposite direction of the translation you configure.
 *If you enable **Bi-directional** translation, it is very important to make sure you have Security policy rules in place to control the traffic in both directions. Without such policy rules, **Bi-directional** translation allows packets to be automatically translated in both directions, which you might not want.*
4. If you want to do destination translation, select **Destination Address Translation**. In the **Translated Address** field, choose an address object or enter your internal destination address.
5. Click **OK**.

STEP 4 | Configure NDP Proxy.

When you configure the firewall to act as an NDP Proxy for addresses, it allows the firewall to send Neighbor Discovery (ND) advertisements and respond to ND solicitations from peers that are asking for MAC addresses of IPv6 prefixes assigned to devices behind the firewall.

1. Select **Network > Interfaces > Ethernet** and select an interface.
2. On the **Advanced > NDP Proxy** tab, select **Enable NDP Proxy** and click **Add**.
3. Enter the **IP Address(es)** for which NDP Proxy is enabled. It can be an address, a range of addresses, or a prefix and prefix length. The order of IP addresses does not matter. These addresses are ideally the same as the Translated Addresses that you configured in an NPTv6 policy.



*If the address is a subnet, the NDP Proxy will respond to all addresses in the subnet, so you should list the neighbors in that subnet with **Negate** selected, as described in the next step.*

4. **(Optional)** Enter one or more addresses for which you do not want NDP Proxy enabled, and select **Negate**. For example, from an IP address range or prefix range configured in the prior step, you could negate a smaller subset of addresses. It is recommended that you negate the addresses of the neighbors of the firewall.

STEP 5 | Commit the configuration.

Click **OK** and **Commit**.

NAT64

NAT64 provides a way to transition to IPv6 while you still need to communicate with IPv4 networks. When you need to communicate from an IPv6-only network to an IPv4 network, you use NAT64 to translate source and destination addresses from IPv6 to IPv4 and vice versa. NAT64 allows IPv6 clients to access IPv4 servers and allows IPv4 clients to access IPv6 servers. You should understand [NAT](#) before configuring NAT64.

- [NAT64 Overview](#)
- [IPv4-Embedded IPv6 Address](#)
- [DNS64 Server](#)
- [Path MTU Discovery](#)
- [IPv6-Initiated Communication](#)
- [Configure NAT64 for IPv6-Initiated Communication](#)
- [Configure NAT64 for IPv4-Initiated Communication](#)
- [Configure NAT64 for IPv4-Initiated Communication with Port Translation](#)

NAT64 Overview

You can configure two types of NAT64 translation on a Palo Alto Networks® firewall; each one is doing a bidirectional translation between the two IP address families:

- The firewall supports stateful NAT64 for [IPv6-Initiated Communication](#), which maps multiple IPv6 addresses to one IPv4 address, thus preserving IPv4 addresses. (It does not support stateless NAT64, which maps one IPv6 address to one IPv4 address and therefore does not preserve IPv4 addresses.) [Configure NAT64 for IPv6-Initiated Communication](#).
- The firewall supports IPv4-initiated communication with a static binding that maps an IPv4 address and port number to an IPv6 address. [Configure NAT64 for IPv4-Initiated Communication](#). It also supports port rewrite, which preserves even more IPv4 addresses by translating an IPv4 address and port number to an IPv6 address with multiple port numbers. [Configure NAT64 for IPv4-Initiated Communication with Port Translation](#).

A single IPv4 address can be used for NAT44 and NAT64; you don't reserve a pool of IPv4 addresses for NAT64 only.

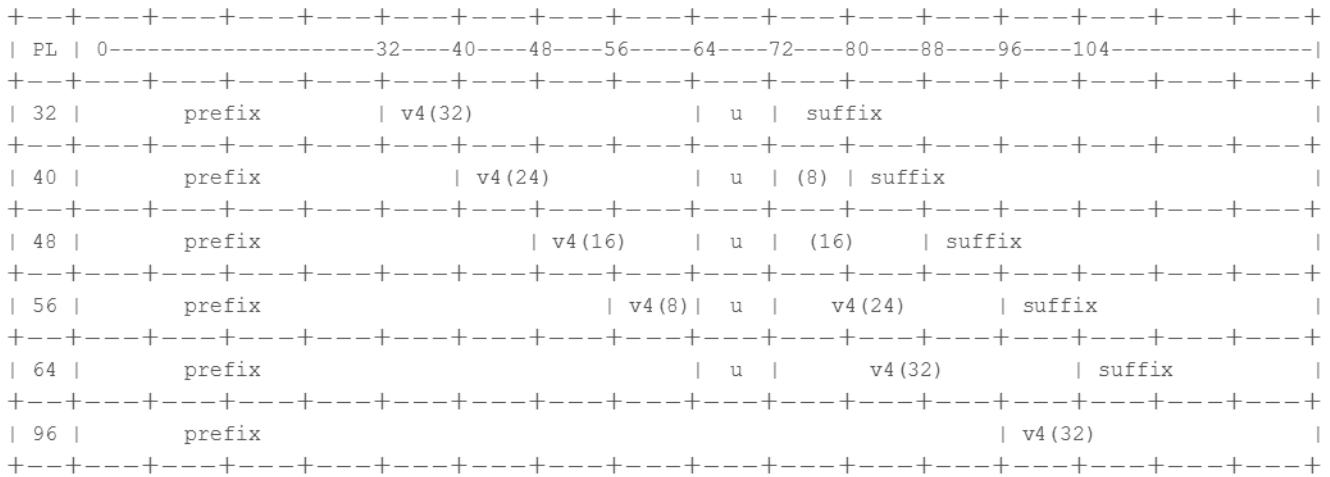
NAT64 operates on Layer 3 interfaces, subinterfaces, and tunnel interfaces. To use NAT64 on a Palo Alto Networks firewall for IPv6-initiated communication, you must have a third-party [DNS64 Server](#) or a solution in place to separate the DNS query function from the NAT function. The DNS64 server translates between your IPv6 host and an IPv4 DNS server by encoding the IPv4 address it receives from a public DNS server into an IPv6 address for the IPv6 host.

Palo Alto Networks supports the following NAT64 features:

- ([PAN-OS 10.2.4 and later 10.2 releases](#)) Beginning with PAN-OS 10.2.4, [persistent NAT for DIPP](#) is available on all firewalls.
- Hairpinning (NAT U-Turn); additionally, NAT64 prevents hairpinning loop attacks by dropping all incoming IPv6 packets that have a source prefix of 64::/n.
- Translation of TCP/UDP/ICMP packets per [RFC 6146](#) and the firewall makes a best effort to translate other protocols that don't use an application-level gateway (ALG). For example, the firewall can translate a GRE packet. This translation has the same limitation as NAT44: if you don't have an ALG for a protocol that can use a separate control and data channel, the firewall might not understand the return traffic flow.
- Translation between IPv4 and IPv6 of the ICMP length attribute of the original datagram field, per [RFC 4884](#).

IPv4-Embedded IPv6 Address

NAT64 uses an IPv4-embedded IPv6 address as described in [RFC 6052, IPv6 Addressing of IPv4/IPv6 Translators](#). An IPv4-embedded IPv6 address is an IPv6 address in which 32 bits have an IPv4 address encoded in them. The IPv6 prefix length (PL in the figure) determines where in the IPv6 address the IPv4 address is encoded, as follows:



The firewall supports translation for /32, /40, /48, /56, /64, and /96 subnets using these prefixes. A single firewall supports multiple prefixes; each NAT64 rule uses one prefix. The prefix can be the Well-Known Prefix (64:FF9B::/96) or a Network-Specific Prefix (NSP) that is unique to the organization that controls the address translator (the DNS64 device). An NSP is usually a network within the organization's IPv6 prefix. The DNS64 device typically sets the u field and suffix to zeros; the firewall ignores those fields.

DNS64 Server

If you need to use a DNS and you want to perform NAT64 translation using [IPv6-Initiated Communication](#), you must use a third-party DNS64 server or other DNS64 solution that is set up with the Well-Known Prefix or your NSP. When an IPv6 host attempts to access an IPv4 host or domain on the internet, the DNS64 server queries an authoritative DNS server for the IPv4 address mapped to that host name. The DNS server returns an Address record (A record) to the DNS64 server containing the IPv4 address for the host name.

The DNS64 server in turn converts the IPv4 address to hexadecimal and encodes it into the appropriate octets of the IPv6 prefix it is set up to use (the Well-Known Prefix or your NSP) based on the prefix length, which results in an [IPv4-Embedded IPv6 Address](#). The DNS64 server sends an AAAA record to the IPv6 host that maps the IPv4-embedded IPv6 address to the IPv4 host name.

Path MTU Discovery

IPv6 does not fragment packets, so the firewall uses two methods to reduce the need to fragment packets:

- When the firewall is translating IPv4 packets in which the DF (don't fragment) bit is zero, that indicates the sender expects the firewall to fragment packets that are too large, but the firewall doesn't fragment packets for the IPv6 network (after translation) because IPv6 doesn't fragment packets. Instead, you can configure the minimum size into which the firewall will fragment IPv4 packets before translating them. The **NAT64 IPv6 Minimum Network MTU** value is this setting, which complies with [RFC 6145, IP/ICMP Translation Algorithm](#). You can set the **NAT64 IPv6 Minimum Network MTU** to its maximum value (**Device > Setup > Session**), which causes the firewall to fragment IPv4 packets to the IPv6 minimum size before translating them to IPv6. (The **NAT64 IPv6 Minimum Network MTU** does not change the interface MTU.)
- The other method the firewall uses to reduce fragmentation is Path MTU Discovery (PMTUD). In an IPv4-initiated communication, if an IPv4 packet to be translated has the DF bit set and the MTU for the egress interface is smaller than the packet, the firewall uses PMTUD to drop the packet and return an ICMP 'Destination Unreachable - fragmentation needed' message to the source. The source lowers the path MTU for that destination and resends the packet until successive reductions in the path MTU allow packet delivery.

IPv6-Initiated Communication

IPv6-initiated communication to the firewall is similar to source NAT for an IPv4 topology.

[Configure NAT64 for IPv6-Initiated Communication](#) when your IPv6 host needs to communicate with an IPv4 server.

In the NAT64 policy rule, configure the original source to be an IPv6 host address or Any. Configure the destination IPv6 address as either the Well-Known Prefix or the NSP that the DNS64 server uses. (You do not configure the full IPv6 destination address in the rule.)

If you need to use a DNS, you need to use a [DNS64 Server](#) to convert an IPv4 DNS “A” result into an “AAAA” result merged with the NAT64 prefix. If you don’t use a DNS, you need to create the address using the IPv4 destination address and the NAT64 prefix configured on the firewall, following [RFC 6052](#) rules.

For environments that use a DNS, the example topology below illustrates communication with the DNS64 server. The DNS64 server must be set up to use the Well-Known Prefix 64:FF9B::/96 or your Network-Specific Prefix, which must comply with RFC 6052 (/32, /40, /48, /56, /64, or /96).

On the translated side of the firewall, the translation type must be Dynamic IP and Port in order to implement stateful NAT64. You configure the source translated address to be the IPv4 address of the egress interface on the firewall. You do not configure the destination translation field; the firewall translates the address by first finding the prefix length in the original destination address of the rule and then based on the prefix, extracting the encoded IPv4 address from the original destination IPv6 address in the incoming packet.

Before the firewall looks at the NAT64 rule, the firewall must do a route lookup to find the destination security zone for an incoming packet. You must ensure that the NAT64 prefix can be reached through the destination zone assignment because the NAT64 prefix should not be routable by the firewall. The firewall would likely assign the NAT64 prefix to the default route or drop the NAT64 prefix because there is no route for it. The firewall will not find a destination zone because the NAT64 prefix is not in its routing table, associated with an egress interface and zone.

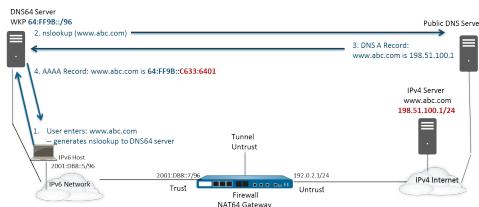
You must also configure a tunnel interface (with no termination point). You apply the NAT64 prefix to the tunnel and apply the appropriate zone to ensure that IPv6 traffic with the NAT64 prefix is assigned to the proper destination zone. The tunnel also has the advantage of dropping IPv6 traffic with the NAT64 prefix if the traffic does not match the NAT64 rule. Your configured routing protocol on the firewall looks up the IPv6 prefix in its routing table to find the destination zone and then looks at the NAT64 rule.

The following figure illustrates the role of the DNS64 server in the name resolution process. In this example, the DNS64 server is configured to use Well-Known Prefix 64:FF9B::/96.

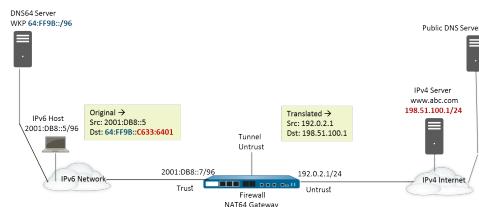
1. A user at the IPv6 host enters the URL www.abc.com, which generates a name server lookup (nslookup) to the DNS64 server.
2. The DNS64 Server sends an nslookup to the public DNS server for www.abc.com, requesting its IPv4 address.
3. The DNS server returns an A record that provides the IPv4 address to the DNS64 server.
4. The DNS64 server sends an AAAA record to the IPv6 user, converting the IPv4 dotted decimal address 198.51.100.1 into C633:6401 hexadecimal and embedding it into its own IPv6

prefix, 64:FF9B::/96. [198 = C6 hex; 51 = 33 hex; 100 = 64 hex; 1 = 01 hex.] The result is **IPv4-Embedded IPv6 Address** 64:FF9B::C633:6401.

Keep in mind that in a /96 prefix, the IPv4 address is the last four octets encoded in the IPv6 address. If the DNS64 server uses a /32, /40, /48, /56 or /64 prefix, the IPv4 address is encoded as shown in RFC 6052.



Upon the transparent name resolution, the IPv6 host sends a packet to the firewall containing its IPv6 source address and destination IPv6 address 64:FF9B::C633:6401 as determined by the DNS64 server. The firewall performs the NAT64 translation based on your NAT64 rule.



Configure NAT64 for IPv6-Initiated Communication

This configuration task and its addresses correspond to the figures in [IPv6-Initiated Communication](#).

Beginning with PAN-OS 10.2.4, you can enable [persistent NAT for DIPP](#) to mitigate the compatibility issues that symmetric NAT may have with applications that use STUN.

STEP 1 | Enable IPv6 to operate on the firewall.

1. Select **Device > Setup > Session** and edit the Session Settings.
2. Select **Enable IPv6 Firewalling**.
3. Click **OK**.

STEP 2 | Configure the interface with IPv6 addressing.

1. Select **Network > Interfaces** and select the interface that performs NAT.
2. Select the **IPv6** tab.
3. Select **Enable IPv6 address on interface**.
4. Add your private IPv6 prefix.
5. Add the well-known prefix `64:ff9b::/96`.
6. Click **OK**.

The screenshot shows the 'Ethernet Interface' configuration window for 'ethernet1/5'. The 'Comment' field is set to 'NAT64'. The 'Interface Type' is 'Layer3' and 'Netflow Profile' is 'None'. The 'IPv6' tab is selected. Under 'Address Assignment', there are two entries: '2001::1/64' and '64:ff9b::/96'. Both addresses are marked as 'ENABLED' with checked checkboxes. The 'Interface ID' is set to 'EUI-64'. At the bottom right are 'OK' and 'Cancel' buttons.

ADDRESS	ENABLED	INTERFACE ID AS HOST	ANYCAST	SEND RA
2001::1/64	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
64:ff9b::/96	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

STEP 3 | Create an address object for the IPv6 destination address (pre-translation).

1. Select **Objects > Addresses** and click **Add**.
2. Enter a **Name** for the object, for example, nat64-IPv4 Server.
3. For **Type**, select **IP Netmask** and enter the IPv6 prefix with a netmask that is compliant with RFC 6052 (/32, /40, /48, /56, /64, or /96). This is either the Well-Known Prefix or your Network-Specific Prefix that is configured on the [DNS64 Server](#).

For this example, enter 64:FF9B::/96.



The source and destination must have the same netmask (prefix length).

(You don't enter a full destination address because, based on the prefix length, the firewall extracts the encoded IPv4 address from the original destination IPv6 address in the incoming packet. In this example, the prefix in the incoming packet is encoded with C633:6401 in hexadecimal, which is the IPv4 destination address 198.51.100.1.)

4. Click **OK**.

STEP 4 | [\(Optional\)](#) Create an address object for the IPv6 source address (pre-translation).

1. Select **Objects > Addresses** and click **Add**.
2. Enter a **Name** for the object.
3. For **Type**, select **IP Netmask** and enter the address of the IPv6 host, in this example, 2001:DB8::5/96.
4. Click **OK**.

STEP 5 | [\(Optional\)](#) Create an address object for the IPv4 source address (translated).

1. Select **Objects > Addresses** and click **Add**.
2. Enter a **Name** for the object.
3. For **Type**, select **IP Netmask** and enter the IPv4 address of the firewall's egress interface, in this example, 192.0.2.1.
4. Click **OK**.

STEP 6 | Create the NAT64 rule.

1. Select **Policies > NAT** and click **Add**.
2. On the **General** tab, enter a **Name** for the NAT64 rule, for example, nat64_ipv6_init.
3. [\(Optional\)](#) Enter a **Description**.
4. For **NAT Type**, select **nat64**.

STEP 7 | Specify the original source and destination information.

1. For the **Original Packet**, **Add the Source Zone**, likely a trusted zone.
2. Select the **Destination Zone**, in this example, the Untrust zone.
3. (**Optional**) Select a **Destination Interface** or the default (**any**).
4. For **Source Address**, select **Any** or **Add** the address object you created for the IPv6 host.
5. For **Destination Address**, **Add** the address object you created for the IPv6 destination address, in this example, nat64-IPv4 Server.
6. (**Optional**) For **Service**, select **any**.

STEP 8 | Specify the translated packet information.

1. For the **Translated Packet**, in **Source Address Translation**, for **Translation Type**, select **Dynamic IP and Port**.
2. For **Address Type**, do one of the following:
 - Select **Translated Address** and **Add** the address object you created for the IPv4 source address.
 - Select **Interface Address**, in which case the translated source address is the IP address and netmask of the firewall's egress interface. For this choice, select an **Interface** and optionally an **IP Address** if the interface has more than one IP address.
3. Leave **Destination Address Translation** unselected. (The firewall extracts the IPv4 address from the IPv6 prefix in the incoming packet, based on the prefix length specified in the original destination of the NAT64 rule.)
4. Click **OK** to save the NAT64 policy rule.

STEP 9 | Configure a tunnel interface to emulate a loopback interface with a netmask other than 128.

1. Select **Network > Interfaces > Tunnel** and **Add** a tunnel.
2. For **Interface Name**, enter a numeric suffix, such as **.2**.
3. On the **Config** tab, select the **Virtual Router** where you are configuring NAT64.
4. For **Security Zone**, select the destination zone associated with the IPv4 server destination (Untrust zone).
5. On the **IPv6** tab, select **Enable IPv6 on the interface**.
6. Click **Add** and for the **Address**, select **New Address**.
7. Enter a **Name** for the address.
8. (**Optional**) Enter a **Description** for the tunnel address.
9. For **Type**, select **IP Netmask** and enter your IPv6 prefix and prefix length, in this example, **64:FF9B::/96**.
10. Click **OK**.
11. Select **Enable address on interface** and click **OK**.
12. Click **OK**.
13. Click **OK** to save the tunnel.

STEP 10 | Create a security policy to allow NAT traffic from the trust zone.

1. Select **Policies > Security** and Add a rule **Name**.
2. Select **Source** and **Add a Source Zone**; select **Trust**.
3. For **Source Address**, select **Any**.
4. Select **Destination** and **Add a Destination Zone**; select **Untrust**.
5. For **Application**, select **Any**.
6. For **Actions**, select **Allow**.
7. Click **OK**.

STEP 11 | Commit your changes.

Click **Commit**.

STEP 12 | (PAN-OS 10.2.4 and later 10.2 releases) Enable persistent NAT for DIPP.

1. [Access the CLI](#).
2. > **set system setting persistent-dipp enable yes**
3. > **request restart system**
4. If you have HA configured, repeat this step on the other HA peer.

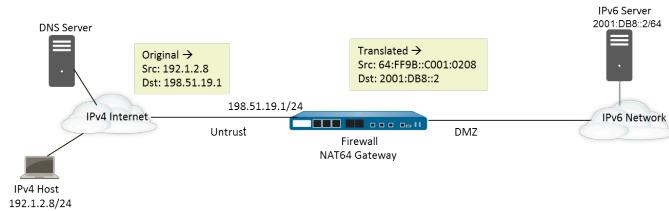
STEP 13 | Troubleshoot or view a NAT64 session.

```
> show session id <session-id>
```

Configure NAT64 for IPv4-Initiated Communication

IPv4-initiated communication to an IPv6 server is similar to destination NAT in an IPv4 topology. The destination IPv4 address maps to the destination IPv6 address through a one-to-one, static IP translation (not a many-to-one translation).

The firewall encodes the source IPv4 address into Well-Known Prefix 64:FF9B::/96 as defined in RFC 6052. The translated destination address is the actual IPv6 address. The use case for IPv4-initiated communication is typically when an organization is providing access from the public, untrust zone to an IPv6 server in the organization's DMZ zone. This topology does not use a DNS64 server.



STEP 1 | Enable IPv6 to operate on the firewall.

1. Select **Device > Setup > Session** and edit the Session Settings.
2. Select **Enable IPv6 Firewalling**.
3. Click **OK**.

STEP 2 | (Optional) When an IPv4 packet has its DF bit set to zero (and because IPv6 does not fragment packets), ensure the translated IPv6 packet does not exceed the path MTU for the destination IPv6 network.

1. Select **Device > Setup > Session** and edit Session Settings.
2. For **NAT64 IPv6 Minimum Network MTU**, enter the smallest number of bytes into which the firewall will fragment IPv4 packets for translation to IPv6 (range is 1280-9216, default is 1280).



If you don't want the firewall to fragment an IPv4 packet prior to translation, set the MTU to 9216. If the translated IPv6 packet still exceeds this value, the firewall drops the packet and issues an ICMP packet indicating destination unreachable - fragmentation needed.

3. Click **OK**.

STEP 3 | Create an address object for the IPv4 destination address (pre-translation).

1. Select **Objects > Addresses** and click **Add**.
2. Enter a **Name** for the object, for example, **nat64_ip4server**.
3. For **Type**, select **IP Netmask** and enter the IPv4 address of the firewall interface in the Untrust zone. The address must use no netmask or a netmask of /32 only. This example uses **198.51.19.1/32**.
4. Click **OK**.

STEP 4 | Create an address object for the IPv6 source address (translated).

1. Select **Objects > Addresses** and click **Add**.
2. Enter a **Name** for the object, for example, nat64_ip6source.
3. For **Type**, select **IP Netmask** and enter the NAT64 IPv6 address with a netmask that is compliant with RFC 6052 (/32, /40, /48, /56, /64, or /96).
For this example, enter 64:FF9B::/96.
(The firewall encodes the prefix with the IPv4 source address 192.1.2.8, which is C001:0208 in hexadecimal.)
4. Click **OK**.

STEP 5 | Create an address object for the IPv6 destination address (translated).

1. Select **Objects > Addresses** and click **Add**.
2. Enter a **Name** for the object, for example, nat64_server_2.
3. For **Type**, select **IP Netmask** and enter the IPv6 address of the IPv6 server (destination).
The address must use no netmask or a netmask of /128 only. This example uses 2001:DB8::2/128.
4. Click **OK**.

STEP 6 | Create the NAT64 rule.

1. Select **Policies > NAT** and click **Add**.
2. On the **General** tab, enter a **Name** for the NAT64 rule, for example, nat64_ipv4_init.
3. For **NAT Type**, select **nat64**.

STEP 7 | Specify the original source and destination information.

1. For the **Original Packet**, **Add the Source Zone**, likely an untrust zone.
2. Select the **Destination Zone**, likely a trust or DMZ zone.
3. For **Source Address**, select **Any** or **Add** the address object for the IPv4 host.
4. For **Destination Address**, **Add** the address object for the IPv4 destination, in this example, nat64_ip4server.
5. For **Service**, select **any**.

STEP 8 | Specify the translated packet information.

1. For the **Translated Packet**, in the **Source Address Translation, Translation Type**, select **Static IP**.
2. For **Translated Address**, select the source translated address object you created, nat64_ip6source.
3. For **Destination Address Translation**, for **Translated Address**, specify a single IPv6 address (the address object, in this example, nat64_server_2, or the IPv6 address of the server).
4. Click **OK**.

STEP 9 | Create a security policy to allow the NAT traffic from the Untrust zone.

1. Select **Policies > Security** and Add a rule **Name**.
2. Select **Source** and **Add a Source Zone**; select **Untrust**.
3. For **Source Address**, select **Any**.
4. Select **Destination** and **Add a Destination Zone**; select **DMZ**.
5. For **Actions**, select **Allow**.
6. Click **OK**.

STEP 10 | Commit your changes.

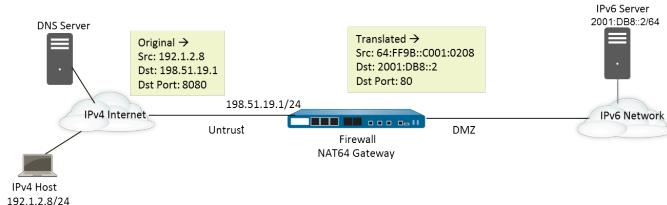
Click **Commit**.

STEP 11 | Troubleshoot or view a NAT64 session.

```
> show session id <session-id>
```

Configure NAT64 for IPv4-Initiated Communication with Port Translation

This task builds on the task to [Configure NAT64 for IPv4-Initiated Communication](#), but the organization controlling the IPv6 network prefers to translate the public destination port number to an internal destination port number and thereby keep it private from users on the IPv4 untrust side of the firewall. In this example, port 8080 is translated to port 80. To do that, in the Original Packet of the NAT64 policy rule, create a new Service that specifies the destination port is 8080. For the Translated Packet, the translated port is 80.



STEP 1 | Enable IPv6 to operate on the firewall.

1. Select **Device > Setup > Session** and edit the Session Settings.
2. Select **Enable IPv6 Firewalling**.
3. Click **OK**.

STEP 2 | (Optional) When an IPv4 packet has its DF bit set to zero (and because IPv6 does not fragment packets), ensure the translated IPv6 packet does not exceed the path MTU for the destination IPv6 network.

1. Select **Device > Setup > Session** and edit Session Settings.
2. For **NAT64 IPv6 Minimum Network MTU**, enter the smallest number of bytes into which the firewall will fragment IPv4 packets for translation to IPv6 (range is 1280-9216, default is 1280).



If you don't want the firewall to fragment an IPv4 packet prior to translation, set the MTU to 9216. If the translated IPv6 packet still exceeds this value, the firewall drops the packet and issues an ICMP packet indicating destination unreachable - fragmentation needed.

3. Click **OK**.

STEP 3 | Create an address object for the IPv4 destination address (pre-translation).

1. Select **Objects > Addresses** and click **Add**.
2. Enter a **Name** for the object, for example, `nat64_ip4server`.
3. For **Type**, select **IP Netmask** and enter the IPv4 address and netmask of the firewall interface in the Untrust zone. This example uses `198.51.19.1/24`.
4. Click **OK**.

STEP 4 | Create an address object for the IPv6 source address (translated).

1. Select **Objects > Addresses** and click **Add**.
2. Enter a **Name** for the object, for example, nat64_ip6source.
3. For **Type**, select **IP Netmask** and enter the NAT64 IPv6 address with a netmask that is compliant with RFC 6052 (/32, /40, /48, /56, /64, or /96).
For this example, enter 64:FF9B::/96.
(The firewall encodes the prefix with the IPv4 source address 192.1.2.8, which is C001:0208 in hexadecimal.)
4. Click **OK**.

STEP 5 | Create an address object for the IPv6 destination address (translated).

1. Select **Objects > Addresses** and click **Add**.
2. Enter a **Name** for the object, for example, nat64_server_2.
3. For **Type**, select **IP Netmask** and enter the IPv6 address of the IPv6 server (destination).
This example uses 2001:DB8::2/64.



The source and destination must have the same netmask (prefix length).

4. Click **OK**.

STEP 6 | Create the NAT64 rule.

1. Select **Policies > NAT** and click **Add**.
2. On the **General** tab, enter a **Name** for the NAT64 rule, for example, nat64_ipv4_init.
3. For **NAT Type**, select **nat64**.

STEP 7 | Specify the original source and destination information, and create a service to limit the translation to a single ingress port number.

1. For the **Original Packet**, Add the **Source Zone**, likely an untrust zone.
2. Select the **Destination Zone**, likely a trust or DMZ zone.
3. For **Service**, select **New Service**.
4. Enter a **Name** for the Service, such as Port_8080.
5. Select **TCP** as the **Protocol**.
6. For **Destination Port**, enter 8080.
7. Click **OK** to save the Service.
8. For **Source Address**, select **Any** or **Add** the address object for the IPv4 host.
9. For **Destination Address**, Add the address object for the IPv4 destination, in this example, nat64_ip4server.

STEP 8 | Specify the translated packet information.

1. For the **Translated Packet**, in the **Source Address Translation**, **Translation Type**, select **Static IP**.
2. For **Translated Address**, select the source translated address object you created, `nat64_ip6source`.
3. For **Destination Address Translation**, for **Translated Address**, specify a single IPv6 address (the address object, in this example, `nat64_server_2`, or the IPv6 address of the server).
4. Specify the private destination **Translated Port** number to which the firewall translates the public destination port number, in this example, 80.
5. Click **OK**.

STEP 9 | Create a security policy to allow the NAT traffic from the Untrust zone.

1. Select **Policies > Security** and Add a rule **Name**.
2. Select **Source** and Add a **Source Zone**; select **Untrust**.
3. For **Source Address**, select **Any**.
4. Select **Destination** and Add a **Destination Zone**; select **DMZ**.
5. For **Actions**, select **Allow**.
6. Click **OK**.

STEP 10 | Commit your changes.

Click **Commit**.

STEP 11 | Troubleshoot or view a NAT64 session.

```
> show session id <session-id>
```


ECMP

Equal Cost Multiple Path (ECMP) processing is a networking feature that enables the firewall to use up to four equal-cost routes to the same destination. Without this feature, if there are multiple equal-cost routes to the same destination, the virtual router chooses one of those routes from the routing table and adds it to its forwarding table; it will not use any of the other routes unless there is an outage in the chosen route.

Enabling ECMP functionality on a virtual router allows the firewall to have up to four equal-cost paths to a destination in its forwarding table, allowing the firewall to:

- Load balance flows (sessions) to the same destination over multiple equal-cost links.
- Efficiently use all available bandwidth on links to the same destination rather than leave some links unused.
- Dynamically shift traffic to another ECMP member to the same destination if a link fails, rather than having to wait for the routing protocol or RIB table to elect an alternative path/route. This can help reduce downtime when links fail.

ECMP is supported on all Palo Alto Networks[®] firewall models, also with hardware forwarding support on the PA-7000 Series, PA-5200 Series, and PA-3200 Series. VM-Series firewalls support ECMP through software only. Performance is affected for sessions that cannot be hardware offloaded.

ECMP is supported on Layer 3, Layer 3 subinterface, VLAN, tunnel, and Aggregated Ethernet interfaces.

ECMP can be configured for static routes and any of the dynamic routing protocols the firewall supports.

ECMP affects the route table capacity because the capacity is based on the number of paths, so an ECMP route with four paths will consume four entries of route table capacity. ECMP implementation might slightly decrease the route table capacity because more memory is being used by session-based tags to map traffic flows to particular interfaces.

ECMP is not supported for equal-cost routes where one or more of those routes has a virtual router or logical router as the next hop. None of the equal-cost routes will be installed in the Forwarding Information Base (FIB).

For information about ECMP path selection when an HA peer fails, see [ECMP in Active/Active HA Mode](#).

The following sections describe ECMP and how to configure it.

- [ECMP Load-Balancing Algorithms](#)
- [Configure ECMP on a Virtual Router](#)
- [Enable ECMP for Multiple BGP Autonomous Systems](#)
- [Verify ECMP](#)

ECMP Load-Balancing Algorithms

Let's suppose the Routing Information Base (RIB) of the firewall has multiple equal-cost paths to a single destination. The maximum number of equal-cost paths defaults to 2. ECMP chooses the best two equal-cost paths from the RIB to copy to the Forwarding Information Base (FIB). ECMP then determines, based on the load-balancing method, which of the two paths in the FIB that the firewall will use for the destination during this session.

ECMP load balancing is done at the session level, not at the packet level—the start of a new session is when the firewall (ECMP) chooses an equal-cost path. The equal-cost paths to a single destination are considered ECMP path members or ECMP group members. ECMP determines which one of the multiple paths to a destination in the FIB to use for an ECMP flow, based on which load-balancing algorithm you set. A virtual router can use only one load-balancing algorithm.

- *Enabling, disabling, or changing ECMP on an existing virtual router causes the system to restart the virtual router, which might cause existing sessions to be terminated.*

The four algorithm choices emphasize different priorities, as follows:

- **Hash-based algorithms prioritize session stickiness**—The **IP Modulo** and **IP Hash** algorithms use hashes based on information in the packet header, such as source and destination address. Because the header of each flow in a given session contains the same source and destination information, these options prioritize session stickiness. If you choose the **IP Hash** algorithm, the hash can be based on the source and destination addresses, or the hash can be based on the source address only. Using an IP hash based on only the source address causes all sessions belonging to the same source IP address to always take the same path from available multiple paths. Thus the path is considered sticky and is easier to troubleshoot if necessary. You can optionally set a **Hash Seed** value to further randomize load balancing if you have a large number of sessions to the same destination and they're not being distributed evenly over the ECMP links.
- **Balanced algorithm prioritizes load balancing**—The **Balanced Round Robin** algorithm distributes incoming sessions equally across the links, favoring load balancing over session stickiness. (Round robin indicates a sequence in which the least recently chosen item is chosen.) In addition, if new routes are added or removed from an ECMP group (for example if a path in the group goes down), the virtual router will re-balance the sessions across links in the group. Additionally, if the flows in a session have to switch routes due to an outage, when the original route associated with the session becomes available again, the flows in the session will revert to the original route when the virtual router once again re-balances the load.
- **Weighted algorithm prioritizes link capacity and/or speed**—As an extension to the ECMP protocol standard, the Palo Alto Networks® implementation provides for a **Weighted Round Robin** load-balancing option that takes into account differing link capacities and speeds on the egress interfaces of the firewall. With this option, you can assign **ECMP Weights** (range is 1 to 255; default is 100) to the interfaces based on link performance using factors such as link capacity, speed, and latency to ensure that loads are balanced to fully leverage the available links.

For example, suppose the firewall has redundant links to an ISP: ethernet1/1 (100 Mbps) and ethernet1/8 (200 Mbps). Although these are equal-cost paths, the link via ethernet1/8

provides greater bandwidth and therefore can handle a greater load than the ethernet1/1 link. Therefore, to ensure that the load-balancing functionality takes into account link capacity and speed, you might assign ethernet1/8 a weight of 200 and ethernet1/1 a weight of 100. The 2:1 weight ratio causes the virtual router to send twice as many sessions to ethernet1/8 as it sends to ethernet1/1. However, because the ECMP protocol is inherently session-based, when using the **Weighted Round Robin** algorithm, the firewall will be able to load balance across the ECMP links only on a best-effort basis.

Keep in mind that ECMP weights are assigned to interfaces to determine load balancing (to influence which *equal-cost* path is chosen), not for route selection (a route choice from routes that could have different costs).



Assign lower-speed or lower-capacity links with a lower weight. Assign higher-speed or higher-capacity links with a higher weight. In this manner, the firewall can distribute sessions based on these ratios, rather than overdrive a low-capacity link that is one of the equal-cost paths.

Configure ECMP on a Virtual Router

Use the following procedure to enable ECMP on a virtual router. The prerequisites are to:

- Specify the interfaces that belong to a virtual router (**Network > Virtual Routers > Router Settings > General**).
- Specify the IP routing protocol.

ECMP is not supported for equal-cost routes where one or more of those routes has a virtual router or logical router as the next hop. None of the equal-cost routes will be installed in the Forwarding Information Base (FIB).

Enabling, disabling, or changing ECMP for an existing virtual router causes the system to restart the virtual router, which might cause sessions to be terminated.

STEP 1 | Enable ECMP for a virtual router.

1. Select **Network > Virtual Routers** and select the virtual router on which to enable ECMP.
2. Select **Router Settings > ECMP** and select **Enable**.

STEP 2 | (Optional) Enable symmetric return of packets from server to client.

Select **Symmetric Return** to cause return packets to egress out the same interface on which the associated ingress packets arrived. That is, the firewall will use the ingress interface on which to send return packets, rather than use the ECMP interface. The **Symmetric Return** setting overrides load balancing. This behavior occurs only for traffic flows from the server to the client.

STEP 3 | Enable **Strict Source Path** to ensure that IKE and IPSec traffic originating at the firewall egresses the physical interface to which the source IP address of the IPSec tunnel belongs.

When you enable ECMP, IKE and IPSec traffic originating at the firewall by default egresses an interface that an ECMP load-balancing method determines. Alternatively, you can ensure that IKE and IPSec traffic originating at the firewall always egresses the physical interface to which the source IP address of the IPSec tunnel belongs, by enabling Strict Source Path. You would enable this function when the firewall has more than one ISP providing equal-cost paths to the same destination. ISPs typically perform a reverse Path Forwarding (RPF) check (or a different check to prevent IP address spoofing) to confirm that traffic is egressing the same interface on which it arrived. Because ECMP would choose an egress interface based on the configured ECMP method (instead of choosing the source interface as the egress interface), that wouldn't be what the ISP expects and the ISP could block legitimate return traffic. In this case, enable Strict Source Path so that the firewall uses the egress interface that is the interface to which the source IP address of the IPSec tunnel belongs, the RPF check succeeds, and the ISP allows the return traffic.

STEP 4 | Specify the maximum number of equal-cost paths (to a destination network) that can be copied from the Routing Information Base (RIB) to the Forwarding Information Base (FIB).

For **Max Path allowed**, enter **2, 3, or 4**. Default: 2.

STEP 5 | Select the load-balancing algorithm for the virtual router. For more information on load-balancing methods and how they differ, see [ECMP Load-Balancing Algorithms](#).

For **Load Balance**, select one of the following options from the **Method** list:

- **IP Modulo** (default)—Uses a hash of the source and destination IP addresses in the packet header to determine which ECMP route to use.
- **IP Hash**—There are two IP hash methods that determine which ECMP route to use (select hash options in Step 5):
 - Use a hash of the source address (available in PAN-OS 8.0.3 and later releases).
 - Use a hash of the source and destination IP addresses (the default IP hash method).
- **Balanced Round Robin**—Uses round robin among the ECMP paths and re-balances paths when the number of paths changes.
- **Weighted Round Robin**—Uses round robin and a relative weight to select from among ECMP paths. Specify the weights in Step 6 below.

STEP 6 | **(IP Hash only)** Configure IP Hash options.

If you selected **IP Hash** as the **Method**:

1. Select **Use Source Address Only** (available in PAN-OS 8.0.3 and later releases) if you want to ensure all sessions belonging to the same source IP address always take the same path from available multiple paths. This IP hash option provides path stickiness and eases troubleshooting. If you don't select this option or you're using a release prior to PAN-OS 8.0.3, the IP hash is based on the source and destination IP addresses (the default IP hash method).



*If you select **Use Source Address Only**, you shouldn't push the configuration from Panorama to firewalls running PAN-OS 8.0.2, 8.0.1, or 8.0.0.*

2. Select **Use Source/Destination Ports** if you want to use source or destination port numbers in the **IP Hash** calculation.



*Enabling this option along with **Use Source Address Only** will randomize path selection even for sessions belonging to the same source IP address.*

3. Enter a **Hash Seed** value (an integer with a maximum of nine digits). Specify a **Hash Seed** value to further randomize load balancing. Specifying a hash seed value is useful if you have a large number of sessions with the same tuple information.

STEP 7 | **(Weighted Round Robin only)** Define a weight for each interface in the ECMP group.

If you selected **Weighted Round Robin** as the **Method**, define a weight for each of the interfaces that are the egress points for traffic to be routed to the same destinations (that is,

interfaces that are part of an ECMP group, such as the interfaces that provide redundant links to your ISP or interfaces to the core business applications on your corporate network).

The higher the weight, the more often that equal-cost path will be selected for a new session.



Give higher speed links a higher weight than a slower links so that more of the ECMP traffic goes over the faster link.

1. Create an ECMP group by clicking **Add** and selecting an **Interface**.
2. **Add** the other interfaces in the ECMP group.
3. Click on **Weight** and specify the relative weight for each interface (range is 1-255; default is 100).

STEP 8 | Save the configuration.

1. Click **OK**.
2. At the ECMP Configuration Change prompt, click **Yes** to restart the virtual router. Restarting the virtual router might cause existing sessions to be terminated.



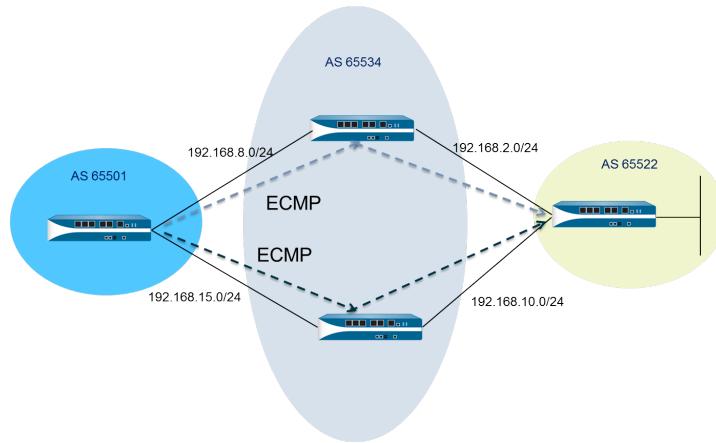
This message displays only if you are modifying an existing virtual router with ECMP.

STEP 9 | Commit your changes.

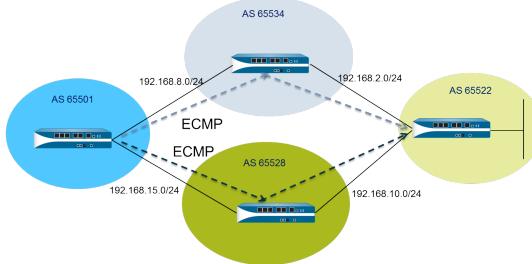
Commit the configuration.

Enable ECMP for Multiple BGP Autonomous Systems

Perform the following task if you have BGP configured, and you want to enable ECMP over multiple autonomous systems. This task presumes that BGP is already configured. In the following figure, two ECMP paths to a destination go through two firewalls belonging to a single ISP in a single BGP autonomous system.



In the following figure, two ECMP paths to a destination go through two firewalls belonging to two different ISPs in different BGP autonomous systems.



STEP 1 | Configure ECMP.

See [Configure ECMP on a Virtual Router](#).

STEP 2 | For BGP routing, enable ECMP over multiple autonomous systems.

1. Select **Network > Virtual Routers** and select the virtual router on which to enable ECMP for multiple BGP autonomous systems.
2. Select **BGP > Advanced** and select **ECMP Multiple AS Support**.

STEP 3 | Commit your changes.

Click **OK** and **Commit**.

Verify ECMP

A virtual router configured for ECMP indicates in the Forwarding Information Base (FIB) table which routes are ECMP routes. An ECMP flag (E) for a route indicates that it is participating in ECMP for the egress interface to the next hop for that route. To verify ECMP, use the following procedure to look at the FIB and confirm that some routes are equal-cost multiple paths.

STEP 1 | Select **Network > Virtual Routers**.

STEP 2 | In the row of the virtual router for which you enabled ECMP, click **More Runtime Stats**.

STEP 3 | Select **Routing > Forwarding Table** to see the FIB.

 *In the table, multiple routes to the same Destination (out a different Interface) have the E flag. An asterisk (*) denotes the preferred path for the ECMP group.*

LLDP

Palo Alto Networks firewalls[®] support Link Layer Discovery Protocol (LLDP), which functions at the link layer to discover neighboring devices and their capabilities. LLDP allows the firewall and other network devices to send and receive LLDP data units (LLDPDUs) to and from neighbors. The receiving device stores the information in a MIB, which the Simple Network Management Protocol (SNMP) can access. LLDP makes troubleshooting easier, especially for virtual wire deployments where the firewall would typically go undetected by a ping or traceroute.

- [LLDP Overview](#)
- [Supported TLVs in LLDP](#)
- [LLDP Syslog Messages and SNMP Traps](#)
- [Configure LLDP](#)
- [View LLDP Settings and Status](#)
- [Clear LLDP Statistics](#)

LLDP Overview

Link Layer Discovery Protocol (LLDP) operates at Layer 2 of the OSI model, using MAC addresses. An LLDPDU is a sequence of type-length-value (TLV) elements encapsulated in an Ethernet frame. The IEEE 802.1AB standard defines three MAC addresses for LLDPDUs: 01-80-C2-00-00-0E, 01-80-C2-00-00-03, and 01-80-C2-00-00-00.

The Palo Alto Networks® firewall supports only one MAC address for transmitting and receiving LLDP data units: 01-80-C2-00-00-0E. When transmitting, the firewall uses 01-80-C2-00-00-0E as the destination MAC address. When receiving, the firewall processes datagrams with 01-80-C2-00-00-0E as the destination MAC address. If the firewall receives either of the other two MAC addresses for LLDPDUs on its interfaces, the firewall takes the same forwarding action it took prior to this feature, as follows:

- If the interface type is vwire, the firewall forwards the datagram to the other port.
- If the interface type is L2, the firewall floods the datagram to the rest of the VLAN.
- If the interface type is L3, the firewall drops the datagrams.

Panorama and the WildFire appliance are not supported.

Interface types that do not support LLDP are tap, high availability (HA), Decrypt Mirror, virtual wire/vlan/L3 subinterfaces, and PA-7000 Series Log Processing Card (LPC) interfaces.

An LLDP Ethernet frame has the following format:

Preamble	Destination MAC	Source MAC	Ethertype	Chassis ID TLV	Port ID TLV	Time To Live TLV	Optional TLVs	End of LLDPDU TLV	Frame Check Sequence
	01:80:C2:00:00:0E or 01:80:C2:00:00:03 or 01:80:C2:00:00:00	Station's Address	0x88CC	Type=1	Type=2	Type=3	Zero or more complete TLVs	Type=0, Length=0	

Within the LLDP Ethernet frame, the TLV structure has the following format:

TLV Type	TLV Information String Length	TLV Information String
7 bits	9 bits	0-511 octets

Supported TLVs in LLDP

LLPDUs include mandatory and optional TLVs. The following table lists the mandatory TLVs that the firewall supports:

Mandatory TLVs	TLV Type	Description
Chassis ID TLV	1	Identifies the firewall chassis. Each firewall must have exactly one unique Chassis ID. The Chassis ID subtype is 4 (MAC address) on Palo Alto Networks® models will use the MAC address of Eth0 to ensure uniqueness.
Port ID TLV	2	Identifies the port from which the LLDPDU is sent. Each firewall uses one Port ID for each LLDPDU message transmitted. The Port ID subtype is 5 (interface name) and uniquely identifies the transmitting port. The firewall uses the interface's ifname as the Port ID.
Time-to-live (TTL) TLV	3	Specifies how long (in seconds) LLDPDU information received from the peer is retained as valid in the local firewall (range is 0-65,535). The value is a multiple of the LLDP Hold Time Multiplier. When the TTL value is 0, the information associated with the device is no longer valid and the firewall removes that entry from the MIB.
End of LLDPDU TLV	0	Indicates the end of the TLVs in the LLDP Ethernet frame.

The following table lists the optional TLVs that the Palo Alto Networks firewall supports:

Optional TLVs	TLV Type	Purpose and Notes Regarding Firewall Implementation
Port Description TLV	4	Describes the port of the firewall in alpha-numeric format. The ifAlias object is used.
System Name TLV	5	Configured name of the firewall in alpha-numeric format. The sysName object is used.
System Description TLV	6	Describes the firewall in alpha-numeric format. The sysDescr object is used.

Optional TLVs	TLV Type	Purpose and Notes Regarding Firewall Implementation
System Capabilities	7	<p>Describes the deployment mode of the interface, as follows:</p> <ul style="list-style-type: none"> An L3 interface is advertised with router (bit 6) capability and the “other” bit (bit 1). An L2 interface is advertised with MAC Bridge (bit 3) capability and the “other” bit (bit 1). A virtual wire interface is advertised with Repeater (bit 2) capability and the “other” bit (bit 1).
Management Address	8	<p>One or more IP addresses used for firewall management, as follows:</p> <ul style="list-style-type: none"> IP address of the management (MGT) interface IPv4 and/or IPv6 address of the interface Loopback address User-defined address entered in the management address field <p>If no management IP address is provided, the default is the MAC address of the transmitting interface.</p> <p>Included is the interface number of the management address specified. Also included is the OID of the hardware interface with the management address specified (if applicable).</p> <p>If more than one management address is specified, they will be sent in the order they are specified, starting at the top of the list. A maximum of four Management Addresses are supported.</p> <p>This is an optional parameter and can be left disabled.</p>

LLDP Syslog Messages and SNMP Traps

The firewall stores LLDP information in MIBs, which an SNMP Manager can monitor. If you want the firewall to send SNMP trap notifications and syslog messages about LLDP events, you must enable **SNMP Syslog Notification** in an LLDP profile.

Per [RFC 5424, The Syslog Protocol](#), and [RFC 1157, A Simple Network Management Protocol](#), LLDP sends syslog and SNMP trap messages when MIB changes occur. These messages are rate-limited by the **Notification Interval**, an LLDP global setting that defaults to 5 seconds and is configurable.

Because the LLDP syslog and SNMP trap messages are rate-limited, some LLDP information provided to those processes might not match the current LLDP statistics seen when you [View the LLDP status information](#). This is normal, expected behavior.

A maximum of 5 MIBs can be received per interface (Ethernet or AE). Each different source has one MIB. If this limit is exceeded, the error message `tooManyNeighbors` is triggered.

Configure LLDP

To configure LLDP and create an LLDP profile, you must be a superuser or device administrator (deviceadmin). A firewall interface supports a maximum of five LLDP peers.

STEP 1 | Enable LLDP on the firewall.

Select **Network > LLDP** and edit the LLDP General section; select **Enable**.

STEP 2 | (Optional) Change LLDP global settings.

1. For **Transmit Interval (sec)**, specify the interval (in seconds) at which LLDPDUs are transmitted. Range is 1 to 3600; default is 30.
2. For **Transmit Delay (sec)**, specify the delay time (in seconds) between LLDP transmissions sent after a change is made in a TLV element. The delay helps to prevent flooding the segment with LLDPDUs if many network changes spike the number of LLDP changes, or if the interface flaps. The **Transmit Delay** must be less than the **Transmit Interval**. Range is 1 to 600; default is 2.
3. For **Hold Time Multiple**, specify a value that is multiplied by the **Transmit Interval** to determine the total TTL Hold Time. Range is 1 to 100; default is 4. The maximum TTL Hold Time is 65535 seconds, regardless of the multiplier value.
4. For **Notification Interval**, specify the interval (in seconds) at which [LLDP Syslog Messages and SNMP Traps](#) are transmitted when MIB changes occur. Range is 1 to 3600; default is 5.
5. Click **OK**.

STEP 3 | Create an LLDP profile.

For descriptions of the optional TLVs, see [Supported TLVs in LLDP](#).

1. Select **Network > Network Profiles > LLDP Profile** and **Add a Name** for the LLDP profile.
2. For **Mode**, select **transmit-receive** (default), **transmit-only**, or **receive-only**.
3. Select **SNMP Syslog Notification** to enable SNMP notifications and syslog messages. If enabled, the global **Notification Interval** is used. The firewall will send both an SNMP trap and a syslog event as configured in the **Device > Log Settings > System > SNMP Trap Profile and Syslog Profile**.
4. For Optional TLVs, select the TLVs you want transmitted:
 - **Port Description**
 - **System Name**
 - **System Description**
 - **System Capabilities**
5. (Optional) Select **Management Address** to add one or more management addresses and **Add a Name**.
6. Select the **Interface** from which to obtain the management address. At least one management address is required if **Management Address** TLV is enabled. If no

management IP address is configured, the system uses the MAC address of the transmitting interface as the management address TLV.

7. Select **IPv4** or **IPv6**, and in the adjacent field, select an IP address from the list (which lists the addresses configured on the selected interface), or enter an address.
8. Click **OK**.
9. Up to four management addresses are allowed. If you specify more than one **Management Address**, they will be sent in the order they are specified, starting at the top of the list. To change the order of the addresses, select an address and use the **Move Up** or **Move Down** buttons.
10. Click **OK**.

STEP 4 | Assign an LLDP profile to an interface.

1. Select **Network > Interfaces** and select the interface where you will assign an LLDP profile.
2. Select **Advanced > LLDP**.
3. Select **Enable LLDP** to assign an LLDP profile to the interface.
4. For **Profile**, select the profile you created. Selecting **None** enables LLDP with basic functionality: sends the three mandatory TLVs and enables **transmit-receive** mode.
If you want to create a new profile, click **LLDP Profile** and follow the instructions steps above.
5. Click **OK**.

STEP 5 | Commit your changes.

View LLDP Settings and Status

Perform the following procedure to view LLDP settings and status.

STEP 1 | View LLDP global settings.

Select Network > LLDP.

On the LLDP General screen, **Enable** indicates whether LLDP is enabled or not.

- If LLDP is enabled, the configured global settings (Transmit Interval, Transmit Delay, Hold Time Multiple, and Notification Interval) are displayed.
- If LLDP is not enabled, the default values of the global settings are displayed.

For descriptions of these values, see second step in [Configure LLDP](#).

STEP 2 | View the LLDP status information.

1. Select the **Status** tab.
2. (**Optional**) Enter a filter to restrict the information that is displayed.

Interface Information:

- **Interface**—Name of the interfaces that have LLDP profiles assigned to them.
- **LLDP**—LLDP status: enabled or disabled.
- **Mode**—LLDP mode of the interface: Tx/Rx, Tx Only, or Rx Only.
- **Profile**—Name of the profile assigned to the interface.

Transmission Information:

- **Total Transmitted**—Count of LLDPDUs transmitted out the interface.
- **Dropped Transmit**—Count of LLDPDUs that were not transmitted out the interface because of an error. For example, a length error when the system is constructing an LLDPDU for transmission.

Received Information:

- **Total Received**—Count of LLDP frames received on the interface.
- **Dropped TLV**—Count of LLDP frames discarded upon receipt.
- **Errors**—Count of TLVs that were received on the interface and contained errors. Types of TLV errors include: one or more mandatory TLVs missing, out of order, containing out-of-range information, or length error.
- **Unrecognized**—Count of TLVs received on the interface that are not recognized by the LLDP local agent. For example, the TLV type is in the reserved TLV range.
- **Aged Out**—Count of items deleted from the Receive MIB due to proper TTL expiration.

STEP 3 | View summary LLDP information for each neighbor seen on an interface.

1. Select the **Peers** tab.
2. (Optional) Enter a filter to restrict the information being displayed.

Local Interface—Interface on the firewall that detected the neighboring device.

Remote Chassis ID—Chassis ID of the peer. The MAC address will be used.

Port ID—Port ID of the peer.

Name—Name of peer.

More info—Provides the following remote peer details, which are based on the Mandatory and Optional TLVs:

- Chassis Type: MAC address.
- MAC Address: MAC address of the peer.
- System Name: Name of the peer.
- System Description: Description of the peer.
- Port Description: Port description of the peer.
- Port Type: Interface name.
- Port ID: The firewall uses the interface's ifname.
- System Capabilities: Capabilities of the system. O=Other, P=Repeater, B=Bridge, W=Wireless-LAN, R=Router, T=Telephone
- Enabled Capabilities: Capabilities enabled on the peer.
- Management Address: Management address of the peer.

Clear LLDP Statistics

You can clear LLDP statistics for specific interfaces.

Clear LLDP statistics for specific interfaces.

1. Select **Network > LLDP > Status** and in the left hand column, select one or more interfaces for which you want to clear LLDP statistics.
2. Click **Clear LLDP Statistics** at the bottom of the screen.

BFD

The firewall supports Bidirectional Forwarding Detection (BFD) ([RFC 5880](#)), a protocol that recognizes a failure in the bidirectional path between two routing peers. BFD failure detection is extremely fast, providing for a faster failover than can be achieved by link monitoring or frequent dynamic routing health checks, such as Hello packets or heartbeats. Mission-critical data centers and networks that require high availability and extremely fast failover need the extremely fast failure detection that BFD provides.

- [BFD Overview](#)
- [Configure BFD](#)
- [Reference: BFD Details](#)

BFD Overview

When you enable BFD, BFD establishes a session from one endpoint (the firewall) to its BFD peer at the endpoint of a link using a three-way handshake. Control packets perform the handshake and negotiate the parameters configured in the BFD profile, including the minimum intervals at which the peers can send and receive control packets. BFD control packets for both IPv4 and IPv6 are transmitted over UDP port 3784. BFD control packets for multihop support are transmitted over UDP port 4784. BFD control packets transmitted over either port are encapsulated in the UDP packets.

After the BFD session is established, the Palo Alto Networks® implementation of BFD operates in asynchronous mode, meaning both endpoints send each other control packets (which function like Hello packets) at the negotiated interval. If a peer does not receive a control packet within the detection time (calculated as the negotiated transmit interval multiplied by a Detection Time Multiplier), the peer considers the session down. (The firewall does not support demand mode, in which control packets are sent only if necessary rather than periodically.)

When you enable BFD for a static route and a BFD session between the firewall and the BFD peer fails, the firewall removes the failed route from the RIB and FIB tables and allows an alternate path with a lower priority to take over. When you enable BFD for a routing protocol, BFD notifies the routing protocol to switch to an alternate path to the peer. Thus, the firewall and BFD peer reconverge on a new path.

A BFD profile allows you to [Configure BFD](#) settings and apply them to one or more routing protocols or static routes on the firewall. If you enable BFD without configuring a profile, the firewall uses its default BFD profile (with all of the default settings). You cannot change the default BFD profile.

When an interface is running multiple protocols that use different BFD profiles, BFD uses the profile having the lowest **Desired Minimum Tx Interval**. See [BFD for Dynamic Routing Protocols](#).

Active/passive HA peers synchronize BFD configurations and sessions; active/active HA peers do not.

BFD is standardized in [RFC 5880](#). PAN-OS does not support all components of RFC 5880; see [Non-Supported RFC Components of BFD](#).

PAN-OS also supports [RFC 5881](#), www.rfc-editor.org/rfc/rfc5881.txt. In this case, BFD tracks a single hop between two systems that use IPv4 or IPv6, so the two systems are directly connected to each other. BFD also tracks multiple hops from peers connected by BGP. PAN-OS follows BFD encapsulation as described in [RFC 5883](#), www.rfc-editor.org/rfc/rfc5883.txt. However, PAN-OS does not support authentication.

- [BFD Model, Interface, and Client Support](#)
- [Non-Supported RFC Components of BFD](#)
- [BFD for Static Routes](#)
- [BFD for Dynamic Routing Protocols](#)

BFD Model, Interface, and Client Support

The following firewall models do not support BFD: PA-800 Series, PA-400 Series, PA-220, and VM-50 firewalls. The models that do support BFD support a maximum number of BFD sessions, as listed in the [Product Selection](#) tool.

BFD runs on physical Ethernet, Aggregated Ethernet (AE), VLAN, and tunnel interfaces (site-to-site VPN and LVPN), and on Layer 3 subinterfaces.

Supported BFD clients are:

- Static routes (IPv4 and IPv6) consisting of a single hop
- OSPFv2 and OSPFv3 (interface types include broadcast, point-to point, and point-to-multipoint)
- BGP IPv4 and IPv6 (IBGP, EBGP) consisting of a single hop or multiple hops
- RIP (single hop)

Non-Supported RFC Components of BFD

- Demand mode
- Authentication
- Sending or receiving Echo packets; however, the firewall will pass Echo packets that arrive on a virtual wire or tap interface. (BFD Echo packets have the same IP address for the source and destination.)
- Poll sequences
- Congestion control
- BFD for LACP (micro-BFD with LAG interfaces)

BFD for Static Routes

To use BFD on a static route, both the firewall and the peer at the opposite end of the static route must support BFD sessions. A static route can have a BFD profile only if the **Next Hop** type is **IP Address**.

If an interface is configured with more than one static route to a peer (the BFD session has the same source IP address and same destination IP address), a single BFD session automatically handles the multiple static routes. This behavior reduces BFD sessions. If the static routes have different BFD profiles, the profile with the smallest **Desired Minimum Tx Interval** takes effect.

In a deployment where you want to configure BFD for a static route on a DHCP or PPPoE client interface, you must perform two commits. Enabling BFD for a static route requires that the **Next Hop** type must be **IP Address**. But at the time of a DHCP or PPPoE interface commit, the interface IP address and next hop IP address (default gateway) are unknown.

You must first enable a DHCP or PPPoE client for the interface, perform a commit, and wait for the DHCP or PPPoE server to send the firewall the client IP address and default gateway IP address. Then you can configure the static route (using the default gateway address of the DHCP or PPPoE client as the next hop), enable BFD, and perform a second commit.

BFD for Dynamic Routing Protocols

In addition to BFD for static routes, the firewall supports BFD for the BGP, OSPF, and RIP routing protocols.



The Palo Alto Networks® implementation of multihop BFD follows the encapsulation portion of [RFC 5883, Bidirectional Forwarding Detection \(BFD\) for Multihop Paths](#) but does not support authentication. A workaround is to configure BFD in a VPN tunnel for BGP. The VPN tunnel can provide authentication without the duplication of BFD authentication.

When you enable BFD for OSPFv2 or OSPFv3 broadcast interfaces, OSPF establishes a BFD session only with its Designated Router (DR) and Backup Designated Router (BDR). On point-to-point interfaces, OSPF establishes a BFD session with the direct neighbor. On point-to-multipoint interfaces, OSPF establishes a BFD session with each peer.

The firewall does not support BFD on an OSPF or OSPFv3 virtual link.

Each routing protocol can have independent BFD sessions on an interface. Alternatively, two or more routing protocols (BGP, OSPF, and RIP) can share a common BFD session for an interface.

When you enable BFD for multiple protocols on the same interface, and the source IP address and destination IP address for the protocols are also the same, the protocols share a single BFD session, thus reducing both dataplane overhead (CPU) and traffic load on the interface. If you configure different BFD profiles for these protocols, only one BFD profile is used: the one that has the lowest **Desired Minimum Tx Interval**. If the profiles have the same **Desired Minimum Tx Interval**, the profile used by the first created session takes effect. In the case where a static route and OSPF share the same session, because a static session is created right after a commit, while OSPF waits until an adjacency is up, the profile of the static route takes effect.

The benefit of using a single BFD session in these cases is that this behavior uses resources more efficiently. The firewall can use the saved resources to support more BFD sessions on different interfaces or support BFD for different source IP and destination IP address pairs.

IPv4 and IPv6 on the same interface always create different BFD sessions, even though they can use the same BFD profile.



If you implement both BFD for BGP and HA path monitoring, Palo Alto Networks recommends you not implement BGP Graceful Restart. When the BFD peer's interface fails and path monitoring fails, BFD can remove the affected routes from the routing table and synchronize this change to the passive HA firewall before Graceful Restart can take effect. If you decide to implement BFD for BGP, Graceful Restart for BGP, and HA path monitoring, you should configure BFD with a larger Desired Minimum Tx Interval and larger Detection Time Multiplier than the default values.

Configure BFD

After you read the [BFD Overview](#), which includes firewall models and interfaces supported, perform the following before configuring BFD:

- Configure one or more [virtual routers](#).
- Configure one or more [Static Routes](#) if you are applying BFD to static routes.
- Configure a routing protocol ([BGP](#), [OSPF](#), [OSPFv3](#), or [RIP](#)) if you are applying BFD to a routing protocol.



The effectiveness of your BFD implementation depends on a variety of factors, such as traffic loads, network conditions, how aggressive your BFD settings are, and how busy the dataplane is.

STEP 1 | Create a BFD profile.



If you change a setting in a BFD profile that an existing BFD session is using and you commit the change, before the firewall deletes that BFD session and recreates it with the new setting, the firewall sends a BFD packet with the local state set to admin down. The peer device may or may not flap the routing protocol or static route, depending on the peer's implementation of [RFC 5882](#), Section 3.2.

1. Select **Network > Network Profiles > BFD Profile** and **Add a Name** for the BFD profile. The name is case-sensitive and must be unique on the firewall. Use only letters, numbers, spaces, hyphens, and underscores.
2. Select the **Mode** in which BFD operates:
 - **Active**—BFD initiates sending control packets to peer (default). At least one of the BFD peers must be Active; both can be Active.
 - **Passive**—BFD waits for peer to send control packets and responds as required.

STEP 2 | Configure BFD intervals.

1. Enter the **Desired Minimum Tx Interval (ms)**. This is the minimum interval, in milliseconds, at which you want the BFD protocol (referred to as BFD) to send BFD control packets; you are thus negotiating the transmit interval with the peer. Minimum on PA-7000 and PA-5200 Series firewalls is 50; minimum on VM-Series firewall is 200. Maximum is 2,000; default is 1,000.



*The recommendation is to set the **Desired Minimum Tx Interval** on a PA-7000 Series firewall to 100 or greater; a value less than 100 is at risk of causing BFD flaps.*



*If you have multiple routing protocols that use different BFD profiles on the same interface, configure the BFD profiles with the same **Desired Minimum Tx Interval**.*

2. Enter the **Required Minimum Rx Interval (ms)**. This is the minimum interval, in milliseconds, at which BFD can receive BFD control packets. Minimum on PA-7000

and PA-5200 Series firewalls is 50; minimum on VM-Series firewall is 200. Maximum is 2,000; default is 1,000.



The recommendation is to set the **Required Minimum Rx Interval** on a PA-7000 Series firewall to 100 or greater; a value less than 100 is at risk of causing BFD flaps.

STEP 3 | Configure the BFD Detection Time Multiplier.

Enter the **Detection Time Multiplier**. The local system calculates the detection time as the **Detection Time Multiplier** received from the remote system multiplied by the agreed transmit interval of the remote system (the greater of the **Required Minimum Rx Interval** and the last received **Desired Minimum Tx Interval**). If BFD does not receive a BFD control packet from its peer before the detection time expires, a failure has occurred. Range is 2 to 50; default is 3.

For example, a transmit interval of 300 ms x 3 (Detection Time Multiplier) = 900 ms detection time.



When configuring a BFD profile, take into consideration that the firewall is a session-based device typically at the edge of a network or data center and may have slower links than a dedicated router. Therefore, the firewall likely needs a longer interval and a higher multiplier than the fastest settings allowed. A detection time that is too short can cause false failure detections when the issue is really just traffic congestion.

STEP 4 | Configure the BFD hold time.

Enter the **Hold Time (ms)**. This is the delay, in milliseconds, after a link comes up before BFD transmits BFD control packets. **Hold Time** applies to BFD Active mode only. If BFD receives BFD control packets during the **Hold Time**, it ignores them. Range is 0-120000. The default is 0, which means no transmit **Hold Time** is used; BFD sends and receives BFD control packets immediately after the link is established.

STEP 5 | (Optional—For a BGP IPv4 implementation only) Configure hop-related settings for the BFD profile.

1. Select **Multihop** to enable BFD over BGP multihop.
2. Enter the **Minimum Rx TTL**. This is the minimum Time-to-Live value (number of hops) BFD will accept (receive) in a BFD control packet when BGP supports multihop BFD. (Range is 1-254; there is no default).

The firewall drops the packet if it receives a smaller TTL than its configured **Minimum Rx TTL**. For example, if the peer is 5 hops away, and the peer transmits a BFD packet with a TTL of 100 to the firewall, and if the **Minimum Rx TTL** for the firewall is set to 96 or higher, the firewall drops the packet.

STEP 6 | Save the BFD profile.

Click **OK**.

STEP 7 | (Optional) Enable BFD for a static route.

Both the firewall and the peer at the opposite end of the static route must support BFD sessions.

1. Select **Network > Virtual Routers** and select the virtual router where the static route is configured.
2. Select the **Static Routes** tab.
3. Select the **IPv4** or **IPv6** tab.
4. Select the static route where you want to apply BFD.
5. Select an **Interface** (even if you are using a DHCP address). The **Interface** setting cannot be **None**.
6. For **Next Hop**, select **IP Address** and enter the IP address if not already specified.
7. For **BFD Profile**, select one of the following:
 - **default**—Uses only default settings.
 - A BFD profile you configured—See [Create a BFD profile](#).
 - **New BFD Profile**—Allows you to [Create a BFD profile](#).



Selecting None (Disable BFD) disables BFD for this static route.

8. Click **OK**.

A BFD column on the **IPv4** or **IPv6** tab indicates the BFD profile configured for the static route.

STEP 8 | (Optional) Enable BFD for all BGP interfaces or for a single BGP peer.

 If you enable or disable BFD globally, all interfaces running BGP will be taken down and brought back up with the BFD function. This can disrupt all BGP traffic. When you enable BFD on the interface, the firewall stops the BGP connection to the peer to program BFD on the interface. The peer device sees the BGP connection drop, which can result in a reconvergence. Enable BFD for BGP interfaces during an off-peak time when a reconvergence will not impact production traffic.

 If you implement both BFD for BGP and HA path monitoring, Palo Alto Networks recommends you not implement BGP Graceful Restart. When the BFD peer's interface fails and path monitoring fails, BFD can remove the affected routes from the routing table and synchronize this change to the passive HA firewall before Graceful Restart can take effect. If you decide to implement BFD for BGP, Graceful Restart for BGP, and HA path monitoring, you should configure BFD with a larger Desired Minimum Tx Interval and larger Detection Time Multiplier than the default values.

1. Select **Network > Virtual Routers** and select the virtual router where BGP is configured.
2. Select the **BGP** tab.
3. (**Optional**) To apply BFD to all BGP interfaces on the virtual router, in the **BFD** list, select one of the following and click **OK**:
 - **default**—Uses only default settings.
 - A BFD profile you configured—See [Create a BFD profile](#).
 - **New BFD Profile**—Allows you to [Create a BFD profile](#).



Selecting **None (Disable BFD)** disables BFD for all BGP interfaces on the virtual router; you cannot enable BFD for a single BGP interface.

4. (**Optional**) To enable BFD for a single BGP peer interface (thereby overriding the **BFD** setting for BGP as long as it is not disabled), perform the following tasks:
 1. Select the **Peer Group** tab.
 2. Select a peer group.
 3. Select a peer.
 4. In the **BFD** list, select one of the following:

default—Uses only default settings.

Inherit-vr-global-setting (default)—The BGP peer inherits the BFD profile that you selected globally for BGP for the virtual router.

A BFD profile you configured—See [Create a BFD profile](#).



Selecting **Disable BFD** disables BFD for the BGP peer.

5. Click **OK**.
6. Click **OK**.

A BFD column on the BGP - Peer Group/Peer list indicates the BFD profile configured for the interface.

STEP 9 | (Optional) Enable BFD for OSPF or OSPFv3 globally or for an OSPF interface.

1. Select **Network > Virtual Routers** and select the virtual router where OSPF or OSPFv3 is configured.
2. Select the **OSPF** or **OSPFv3** tab.
3. (Optional) In the **BFD** list, select one of the following to enable BFD for all OSPF or OSPFv3 interfaces and click **OK**:
 - **default**—Uses only default settings.
 - A BFD profile you configured—See [Create a BFD profile](#).
 - **New BFD Profile**—Allows you to [Create a BFD profile](#).



*Selecting **None (Disable BFD)** disables BFD for all OSPF interfaces on the virtual router; you cannot enable BFD for a single OSPF interface.*

4. (Optional) To enable BFD on a single OSPF peer interface (and thereby override the **BFD** setting for OSPF, as long as it is not disabled), perform the following tasks:
 1. Select the **Areas** tab and select an area.
 2. On the **Interface** tab, select an interface.
 3. In the **BFD** list, select one of the following to configure BFD for the specified OSPF peer:
 - default**—Uses only default settings.
 - Inherit-vr-global-setting** (default)—OSPF peer inherits the **BFD** setting for OSPF or OSPFv3 for the virtual router.
 - A BFD profile you configured—See [Create a BFD profile](#).



*Selecting **Disable BFD** disables BFD for the OSPF or OSPFv3 interface.*

4. Click **OK**.
5. Click **OK**.

A BFD column on the **OSPF Interface** tab indicates the BFD profile configured for the interface.

STEP 10 | (Optional) Enable BFD for RIP globally or for a single RIP interface.

1. Select **Network > Virtual Routers** and select the virtual router where RIP is configured.
2. Select the **RIP** tab.
3. (**Optional**) In the **BFD** list, select one of the following to enable BFD for all RIP interfaces on the virtual router and click **OK**:
 - **default**—Uses only default settings.
 - A BFD profile you configured—See [Create a BFD profile](#).
 - **New BFD Profile**—Allows you to [Create a BFD profile](#).



Selecting None (Disable BFD) disables BFD for all RIP interfaces on the virtual router; you cannot enable BFD for a single RIP interface.

4. (**Optional**) To enable BFD for a single RIP interface (and thereby override the **BFD** setting for RIP, as long as it is not disabled), perform the following tasks:
 1. Select the **Interfaces** tab and select an interface.
 2. In the **BFD** list, select one of the following:
 - default**—Uses only default settings).
 - Inherit-vr-global-setting** (default)—RIP interface inherits the BFD profile that you selected for RIP globally for the virtual router.
 - A BFD profile you configured—See [Create a BFD profile](#).



Selecting None (Disable BFD) disables BFD for the RIP interface.

3. Click **OK**.
5. Click **OK**.

The BFD column on the **Interface** tab indicates the BFD profile configured for the interface.

STEP 11 | Commit the configuration.

Click **Commit**.

STEP 12 | View BFD summary and details.

1. Select **Network > Virtual Routers**, find the virtual router you are interested in, and click **More Runtime Stats**.
2. Select the **BFD Summary Information** tab to see summary information, such as BFD state and run-time statistics.
3. (**Optional**) Select **details** in the row of the interface you are interested in to view Reference: [BFD Details](#).

STEP 13 | Monitor BFD profiles referenced by a routing configuration; monitor BFD statistics, status, and state.

Use the following CLI operational commands:

- **show routing bfd active-profile [<name>]**
- **show routing bfd details [interface<name>][local-ip<ip>][multihop][peer-ip <ip>][session-id][virtual-router<name>]**
- **show routing bfd drop-counters session-id <session-id>**
- **show counter global | match bfd**

STEP 14 | (Optional) Clear BFD transmit, receive, and drop counters.

```
clear routing bfd counters session-id all | <1-1024>
```

STEP 15 | (Optional) Clear BFD sessions for debugging.

```
clear routing bfd session-state session-id all | <1-1024>
```

Reference: BFD Details

To see the following BFD information for a virtual router, refer to Step 12 of [Configure BFD](#), View BFD summary and details.

Name	Value (Example)	Description
Session ID	1	ID number of the BFD session.
Interface	ethernet1/12	Interface you selected where BFD is running.
Protocol	STATIC(IPV4) OSPF	Static route (IP address family of static route) and/or dynamic routing protocol that is running BFD on the interface.
Local IP Address	10.55.55.2	IP address of interface.
Neighbor IP Address	10.55.55.1	IP address of BFD neighbor.
BFD Profile	default *(This BFD session has multiple BFD profiles. Lowest 'Desired Minimum Tx Interval (ms)' is used to select the effective profile.)	Name of BFD profile applied to the interface. Because the sample interface has both a static route and OSPF running BFD with different profiles, the firewall uses the profile with the lowest Desired Minimum Tx Interval . In this example, the profile used is the default profile.
State (local/remote)	up/up	BFD states of the local and remote BFD peers. Possible states are admin down, down, init, and up.
Up Time	2h 36m 21s 419ms	Length of time BFD has been up (hours, minutes, seconds, and milliseconds).
Discriminator (local/remote)	1391591427/1	Discriminators for local and remote BFD peers.
Mode	Active	Mode in which BFD is configured on the interface: Active or Passive.
Demand Mode	Disabled	PAN-OS does not support BFD Demand Mode, so it is always in Disabled state.
Multihop	Disabled	BFD multihop: Enabled or Disabled.

Name	Value (Example)	Description
Multihop TTL		TTL of multihop; range is 1-254. Field is empty if Multihop is disabled.
Local Diag Code	0 (No Diagnostic)	<p>Diagnostic codes indicating the reason for the local system's last change in state:</p> <ul style="list-style-type: none"> 0—No Diagnostic 1—Control Detection Time Expired 2—Echo Function Failed 3—Neighbor Signaled Session Down 4—Forwarding Plane Reset 5—Path Down 6—Concatenated Path Down 7—Administratively Down 8—Reverse Concatenated Path Down
Last Received Remote Diag Code	0 (No Diagnostic)	Diagnostic code last received from BFD peer.
Transmit Hold Time	0ms	Hold time (in milliseconds) after a link comes up before BFD transmits BFD control packets. A hold time of 0ms means to transmit immediately. Range is 0-120000ms.
Received Min Rx Interval	1000ms	Minimum Rx interval received from the peer; the interval at which the BFD peer can receive control packets. Maximum is 2000ms.
Negotiated Transmit Interval	1000ms	Transmit interval (in milliseconds) that the BFD peers have agreed to send BFD control packets to each other. Maximum is 2000ms.
Received Multiplier	3	Detection time multiplier value received from the BFD peer. The Transmit Time multiplied by the Multiplier equals the detection time. If BFD does not receive a BFD control packet from its peer before the detection time expires, a failure has occurred. Range is 2-50.
Detect Time (exceeded)	3000ms (0)	Calculated detection time (Negotiated Transmit Interval multiplied by Multiplier) and the number of milliseconds the detection time is exceeded.

Name	Value (Example)	Description
Tx Control Packets (last)	9383 (420ms ago)	Number of BFD control packets transmitted (and length of time since BFD transmitted the most recent control packet).
Rx Control Packets (last)	9384 (407ms ago)	Number of BFD control packets received (and length of time since BFD received the most recent control packet).
Agent Data Plane	Slot 1 - DP 0	On PA-7000 Series firewalls, the dataplane CPU that is assigned to handle packets for this BFD session.
Errors	0	Number of BFD errors.

Last Packet Causing State Change

Version	1	BFD version.
Poll Bit	0	BFD poll bit; 0 indicates not set.
Desired Min Tx Interval	1000ms	Desired minimum transmit interval of last packet causing state change.
Required Min Rx Interval	1000ms	Required minimum receive interval of last packet causing state change.
Detect Multiplier	3	Detect Multiplier of last packet causing state change.
My Discriminator	1	Remote discriminator. A discriminator is a unique, nonzero value the peers use to distinguish multiple BFD sessions between them.
Your Discriminator	1391591427	Local discriminator. A discriminator is a unique, nonzero value the peers use to distinguish multiple BFD sessions between them.
Diagnostic Code	0 (No Diagnostic)	Diagnostic code of last packet causing state change.
Length	24	Length of BFD control packet in bytes.
Demand Bit	0	PAN-OS does not support BFD Demand mode, so Demand Bit is always set to 0 (disabled).

Name	Value (Example)	Description
Final Bit	0	PAN-OS does not support the Poll Sequence, so Final Bit is always set to 0 (disabled).
Multipoint Bit	0	This bit is reserved for future point-to-multipoint extensions to BFD. It must be zero on both transmit and receipt.
Control Plane Independent Bit	1	<ul style="list-style-type: none"> If set to 1, the transmitting system's BFD implementation does not share fate with its control plane (i.e., BFD is implemented in the forwarding plane and can continue to function through disruptions in the control plane). In PAN-OS, this bit is always set to 1. If set to 0, the transmitting system's BFD implementation shares fate with its control plane.
Authentication Present Bit	0	PAN-OS does not support BFD Authentication, so the Authentication Present Bit is always set to 0.
Required Min Echo Rx Interval	0ms	PAN-OS does not support the BFD Echo function, so this will always be 0ms.

Session Settings and Timeouts

This section describes the global settings that affect TCP, UDP, and ICMPv6 sessions, in addition to IPv6, NAT64, NAT oversubscription, jumbo frame size, MTU, accelerated aging, and Captive Portal authentication. There is also a setting (Rematch Sessions) that allows you to apply newly configured security policies to sessions that are already in progress.

The first few topics below provide brief summaries of the Transport Layer of the OSI model, TCP, UDP, and ICMP. For more information about the protocols, refer to their respective RFCs. The remaining topics describe the session timeouts and settings.

- [Transport Layer Sessions](#)
- [TCP](#)
- [UDP](#)
- [ICMP](#)
- [Control Specific ICMP or ICMPv6 Types and Codes](#)
- [Configure Session Timeouts](#)
- [Session Distribution Policies](#)
- [Configure Session Settings](#)
- [Prevent TCP Split Handshake Session Establishment](#)

Transport Layer Sessions

A network session is an exchange of messages that occurs between two or more communication devices, lasting for some period of time. A session is established and is torn down when the session ends. Different types of sessions occur at three layers of the OSI model: the Transport layer, the Session layer, and the Application layer.

The Transport Layer operates at Layer 4 of the OSI model, providing reliable or unreliable, end-to-end delivery and flow control of data. Internet protocols that implement sessions at the Transport layer include Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

TCP

Transmission Control Protocol (TCP) ([RFC 793](#)) is one of the main protocols in the Internet Protocol (IP) suite, and is so prevalent that it is frequently referenced together with IP as TCP/IP. TCP is considered a reliable transport protocol because it provides error-checking while transmitting and receiving segments, acknowledges segments received, and reorders segments that arrive in the wrong order. TCP also requests and provides retransmission of segments that were dropped. TCP is stateful and connection-oriented, meaning a connection between the sender and receiver is established for the duration of the session. TCP provides flow control of packets, so it can handle congestion over networks.

TCP performs a handshake during session setup to initiate and acknowledge a session. After the data is transferred, the session is closed in an orderly manner, where each side transmits a FIN packet and acknowledges it with an ACK packet. The handshake that initiates the TCP session is often a three-way handshake (an exchange of three messages) between the initiator and the listener, or it could be a variation, such as a four-way or five-way split handshake or a simultaneous open. The [TCP Split Handshake Drop](#) explains how to [Prevent TCP Split Handshake Session Establishment](#).

Applications that use TCP as their transport protocol include Hypertext Transfer Protocol (HTTP), HTTP Secure (HTTPS), File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), Telnet, Post Office Protocol version 3 (POP3), Internet Message Access Protocol (IMAP), and Secure Shell (SSH).

The following topics describe details of the PAN-OS implementation of TCP.

- [TCP Half Closed and TCP Time Wait Timers](#)
- [Unverified RST Timer](#)
- [TCP Split Handshake Drop](#)
- [Maximum Segment Size \(MSS\)](#)

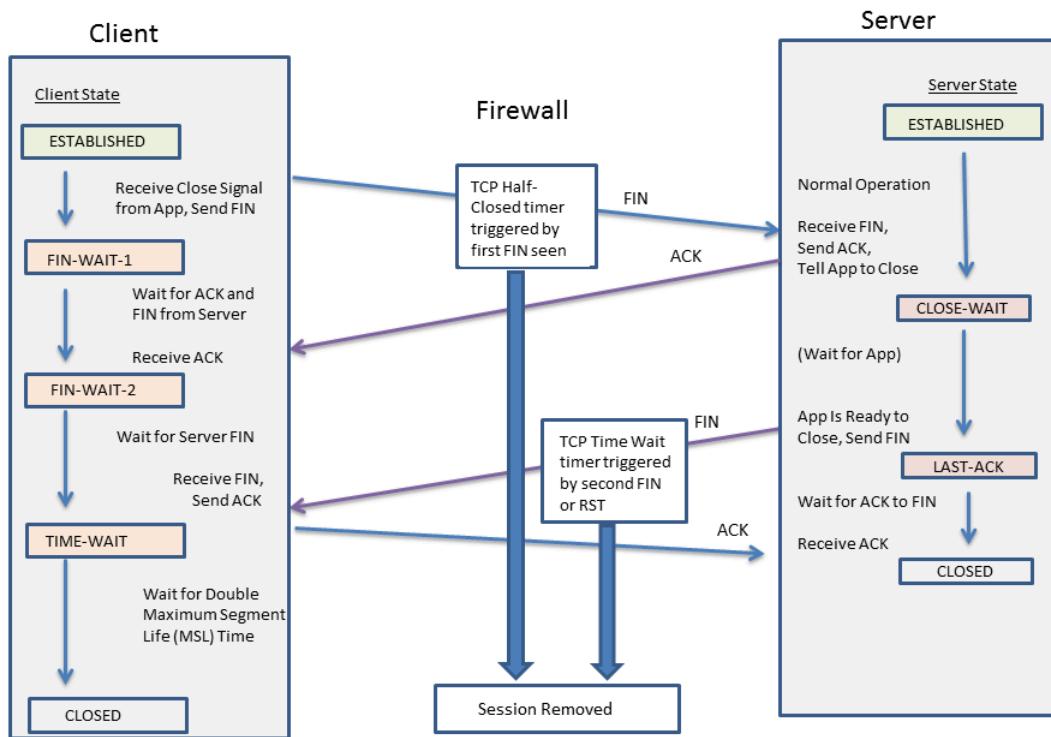
You can configure [packet-based attack protection](#) and thereby drop IP, TCP, and IPv6 packets with undesirable characteristics or strip undesirable options from packets before allowing them into the zone. You can also configure flood protection, specifying the rate of SYN connections per second (not matching an existing session) that trigger an alarm, cause the firewall to randomly drop SYN packets or use SYN cookies, and cause the firewall to drop SYN packets that exceed the maximum rate.

TCP Half Closed and TCP Time Wait Timers

The TCP connection termination procedure uses a TCP Half Closed timer, which is triggered by the first FIN the firewall sees for a session. The timer is named TCP Half Closed because only one side of the connection has sent a FIN. A second timer, TCP Time Wait, is triggered by the second FIN or a RST.

If the firewall were to have only one timer triggered by the first FIN, a setting that was too short could prematurely close the half-closed sessions. Conversely, a setting that was too long would make the session table grow too much and possibly use up all of the sessions. Two timers allow you to have a relatively long TCP Half Closed timer and a short TCP Time Wait timer, thereby quickly aging fully closed sessions and controlling the size of the session table.

The following figure illustrates when the firewall's two timers are triggered during the TCP connection termination procedure.



The TCP Time Wait timer should be set to a value less than the TCP Half Closed timer for the following reasons:

- The longer time allowed after the first FIN is seen gives the opposite side of the connection time to fully close the session.
- The shorter Time Wait time is because there is no need for the session to remain open for a long time after the second FIN or a RST is seen. A shorter Time Wait time frees up resources sooner, yet still allows time for the firewall to see the final ACK and possible retransmission of other datagrams.

If you configure a TCP Time Wait timer to a value greater than the TCP Half Closed timer, the commit will be accepted, but in practice the TCP Time Wait timer will not exceed the TCP Half Closed value.

The timers can be set globally or per application. The global settings are used for all applications by default. If you configure TCP wait timers at the application level, they override the global settings.

Unverified RST Timer

If the firewall receives a Reset (RST) packet that cannot be verified (because it has an unexpected sequence number within the TCP window or it is from an asymmetric path), the Unverified RST timer controls the aging out of the session. It defaults to 30 seconds; the range is 1-600 seconds.

The Unverified RST timer provides an additional security measure, explained in the second bullet below.

A RST packet will have one of three possible outcomes:

- A RST packet that falls outside the TCP window is dropped.
- A RST packet that falls inside the TCP window but does not have the exact expected sequence number is unverified and subject to the Unverified RST timer setting. This behavior helps prevent denial of service (DoS) attacks where the attack tries to disrupt existing sessions by sending random RST packets to the firewall.
- A RST packet that falls within the TCP window and has the exact expected sequence number is subject to the TCP Time Wait timer setting.

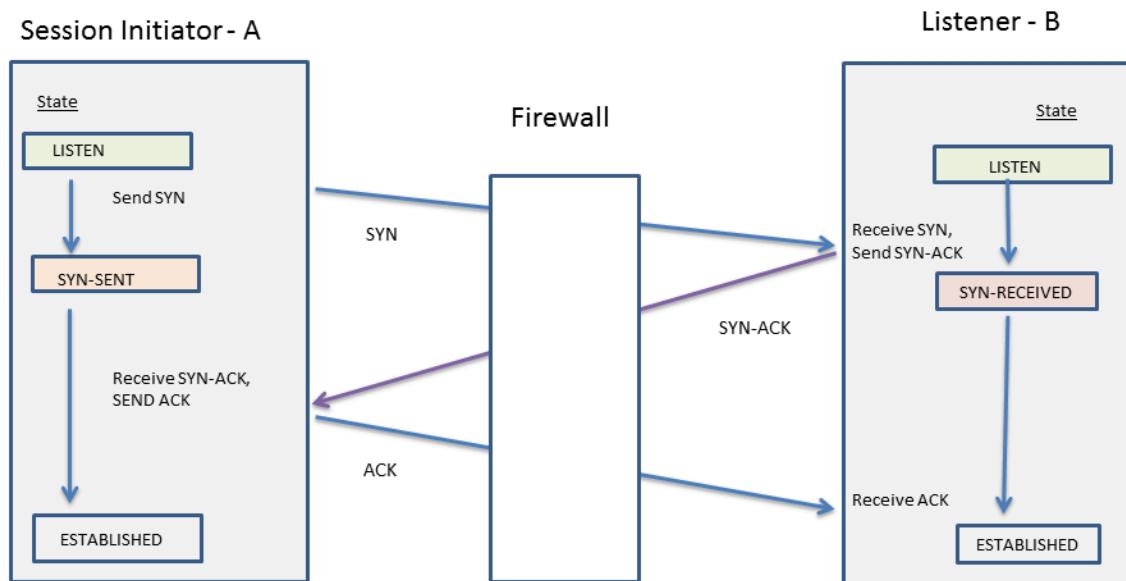
TCP Split Handshake Drop

The **Split Handshake** option in a Zone Protection profile will prevent a TCP session from being established if the session establishment procedure does not use the well-known three-way handshake, but instead uses a variation, such as a four-way or five-way split handshake or a simultaneous open.

The Palo Alto Networks® next-generation firewall correctly handles sessions and all Layer 7 processes for split handshake and simultaneous open session establishment without enabling the **Split Handshake** option. Nevertheless, the **Split Handshake** option (which causes a TCP split handshake drop) is made available. When the **Split Handshake** option is configured for a Zone Protection profile and that profile is applied to a zone, TCP sessions for interfaces in that zone must be established using the standard three-way handshake; variations are not allowed.

The **Split Handshake** option is disabled by default.

The following illustrates the standard three-way handshake used to establish a TCP session with a PAN-OS firewall between the initiator (typically a client) and the listener (typically a server).



The **Split Handshake** option is configured for a Zone Protection profile that is assigned to a zone. An interface that is a member of the zone drops any synchronization (SYN) packets sent from the

server, preventing the following variations of handshakes. The letter A in the figure indicates the session initiator and B indicates the listener. Each numbered segment of the handshake has an arrow indicating the direction of the segment from the sender to the receiver, and each segment indicates the control bit(s) setting.

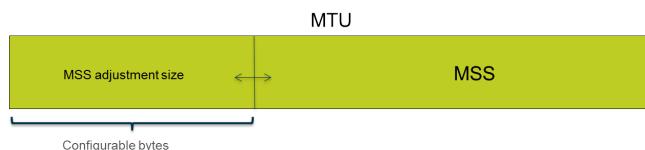
4-Way Split Handshake (Version 1)	4-Way Split Handshake (Version 2)	Simultaneous Open	5-Way Split Handshake
1. A → B SYN 2. A ← B ACK 3. A ← B SYN 4. A → B ACK	1. A → B SYN 2. A ← B SYN 3. A → B SYN-ACK 4. A ← B ACK	1. A → B SYN 2. A ← B SYN 3. A → B SYN-ACK 4. A ← B SYN-ACK	1. A → B SYN 2. A ← B ACK 3. A ← B SYN 4. A → B SYN-ACK 5. A ← B ACK

You can [Prevent TCP Split Handshake Session Establishment](#).

Maximum Segment Size (MSS)

The maximum transmission unit (MTU) is a value indicating the largest number of bytes that can be transmitted in a single TCP packet. The MTU includes the length of headers, so the MTU minus the number of bytes in the headers equals the maximum segment size (MSS), which is the maximum number of data bytes that can be transmitted in a single packet.

A configurable MSS adjustment size (shown below) allows your firewall to pass traffic that has longer headers than the default setting allows. Encapsulation adds length to headers, so you would increase the MSS adjustment size to allow bytes, for example, to accommodate an MPLS header or tunneled traffic that has a VLAN tag.



If the DF (don't fragment) bit is set for a packet, it is especially helpful to have a larger MSS adjustment size and smaller MSS so that longer headers do not result in a packet length that exceeds the allowed MTU. If the DF bit were set and the MTU were exceeded, the larger packets would be dropped.



You can configure the firewall globally to fragment IPv4 packets that exceed the egress interface MTU, even when the DF bit is set in the packet. Enable this for Layer 3 physical interfaces and IPSec tunnel interfaces using the CLI command **debug dataplane set ip4-df-ignore yes**. Restore the firewall to the default behavior by using the CLI command **debug dataplane set ipv4-df-ignore no**.

The firewall supports a configurable MSS adjustment size for IPv4 and IPv6 addresses on the following Layer 3 interface types: Ethernet, subinterfaces, Aggregated Ethernet (AE), VLAN, and loopback. The IPv6 MSS adjustment size applies only if IPv6 is enabled on the interface.



If IPv4 and IPv6 are enabled on an interface and the MSS Adjustment Size differs between the two IP address formats, the proper MSS value corresponding to the IP type is used for TCP traffic.

For IPv4 and IPv6 addresses, the firewall accommodates larger-than-expected TCP header lengths. In the case where a TCP packet has a larger header length than you planned for, the firewall chooses as the MSS adjustment size the larger of the following two values:

- The configured MSS adjustment size
- The sum of the length of the TCP header (20) + the length of IP headers in the TCP SYN

This behavior means that the firewall overrides the configured MSS adjustment size if necessary. For example, if you configure an MSS adjustment size of 42, you expect the MSS to equal 1458 (the default MTU size minus the adjustment size [1500 - 42]). However, the TCP packet has 4 extra bytes of IP options in the header, so the MSS adjustment size (20+20+4) equals 44, which is larger than the configured MSS adjustment size of 42. The resulting MSS is 1500-44=1456 bytes, smaller than you expected.

To configure the MSS adjustment size, see [Configure Session Settings](#).

UDP

User Datagram Protocol (UDP) ([RFC 768](#)) is another main protocol of the IP suite, and is an alternative to TCP. UDP is stateless and connectionless in that there is no handshake to set up a session, and no connection between the sender and receiver; the packets may take different routes to get to a single destination. UDP is considered an unreliable protocol because it does not provide acknowledgments, error-checking, retransmission, or reordering of datagrams. Without the overhead required to provide those features, UDP has reduced latency and is faster than TCP. UDP is referred to as a best-effort protocol because there is no mechanism or guarantee to ensure that the data will arrive at its destination.

A UDP datagram is encapsulated in an IP packet. Although UDP uses a checksum for data integrity, it performs no error checking at the network interface level. Error checking is assumed to be unnecessary or is performed by the application rather than UDP itself. UDP has no mechanism to handle flow control of packets.

UDP is often used for applications that require faster speeds and time-sensitive, real-time delivery, such as Voice over IP (VoIP), streaming audio and video, and online games. UDP is transaction-oriented, so it is also used for applications that respond to small queries from many clients, such as Domain Name System (DNS) and Trivial File Transfer Protocol (TFTP).

You can use Zone Protection Profiles on the firewall to configure [flood protection](#) and thereby specify the rate of UDP connections per second (not matching an existing session) that trigger an alarm, trigger the firewall to randomly drop UDP packets, and cause the firewall to drop UDP packets that exceed the maximum rate. (Although UDP is connectionless, the firewall tracks UDP datagrams in IP packets on a session basis; therefore if the UDP packet doesn't match an existing session, it is considered a new session and it counts as a connection toward the thresholds.)

ICMP

Internet Control Message Protocol (ICMP) ([RFC 792](#)) is another one of the main protocols of the Internet Protocol suite; it operates at the Network layer of the OSI model. ICMP is used for diagnostic and control purposes, to send error messages about IP operations, or messages about requested services or the reachability of a host or router. Network utilities such as traceroute and ping are implemented by using various ICMP messages.

ICMP is a connectionless protocol that does not open or maintain actual sessions. However, the ICMP messages between two devices can be considered a session.

Palo Alto Networks® firewalls support ICMPv4 and ICMPv6. You can control ICMPv4 and ICMPv6 packets in several ways:

- Create [Security Policy Rules Based on ICMP and ICMPv6 Packets](#) and select the **icmp** or **ipv6-icmp** application in the rule.
- Control [ICMPv6 Rate Limiting](#) when you [Configure Session Settings](#).
- Configure [Flood Protection](#), specifying the rate of ICMP or ICMPv6 connections per second (not matching an existing session) that trigger an alarm, trigger the firewall to randomly drop ICMP or ICMPv6 packets, and cause the firewall to drop ICMP or ICMPv6 packets that exceed the maximum rate.
- Configure [Packet-Based Attack Protection](#) packet based attack protection:
 - For ICMP, you can drop certain types of packets or suppress the sending of certain packets.
 - For ICMPv6 packets (Types 1, 2, 3, 4, and 137), you can specify that the firewall use the ICMP session key to match a security policy rule, which determines whether the ICMPv6 packet is allowed or not. (The firewall uses the security policy rule, overriding the default behavior of using the embedded packet to determine a session match.) When the firewall drops ICMPv6 packets that match a security policy rule, the firewall logs the details in Traffic logs.

Security Policy Rules Based on ICMP and ICMPv6 Packets

The firewall forwards ICMP or ICMPv6 packets only if a security policy rule allows the session (as the firewall does for other packet types). The firewall determines a session match in one of two ways, depending on whether the packet is an ICMP or ICMPv6 error packet or redirect packet as opposed to an ICMP or ICMPv6 informational packet:

- **ICMP Types 3, 5, 11, and 12 and ICMPv6 Types 1, 2, 3, 4, and 137**—The firewall by default looks up the embedded IP packet bytes of information from the original datagram that caused the error (the invoking packet). If the embedded packet matches an existing session, the firewall forwards or drops the ICMP or ICMPv6 packet according to the action specified in the security policy rule that matches that same session. (You can use [Packet-Based Attack Protection](#) to override this default behavior for the ICMPv6 types.)
- **Remaining ICMP or ICMPv6 Packet Types**—The firewall treats the ICMP or ICMPv6 packet as if it belongs to a new session. If a security policy rule matches the packet (which the firewall

recognizes as an **icmp** or **ipv6-icmp** session), the firewall forwards or drops the packet based on the security policy rule action. Security policy counters and traffic logs reflect the actions.

If no security policy rule matches the packet, the firewall applies its default security policy rules, which allow intrazone traffic and block interzone traffic (logging is disabled by default for these rules).



Although you can override the default rules to enable logging or change the default action, we don't recommend you change the default behavior for a specific case because it will impact all traffic that those default rules affect. Instead, create security policy rules to control and log ICMP or ICMPv6 packets explicitly.

There are two ways to create explicit security policy rules to handle ICMP or ICMPv6 packets that are not error or redirect packets:

- **Create a security policy rule to allow (or deny) all ICMP or ICMPv6 packets**—In the security policy rule, specify the application **icmp** or **ipv6-icmp**; the firewall allows (or denies) all IP packets matching the ICMP protocol number (1) or ICMPv6 protocol number (58), respectively, through the firewall.
- **Create a custom application and a security policy rule to allow (or deny) packets from or to that application**—This more granular approach allows you to [Control Specific ICMP or ICMPv6 Types and Codes](#).

ICMPv6 Rate Limiting

ICMPv6 rate limiting is a throttling mechanism to prevent flooding and DDoS attempts. The implementation employs an error packet rate and a token bucket, which work together to enable throttling and ensure that ICMP packets don't flood the network segments protected by the firewall.

First the global **ICMPv6 Error Packet Rate (per sec)** controls the rate at which ICMPv6 error packets are allowed through the firewall; the default is 100 packets per second; the range is 10 to 65535 packets per second. If the firewall reaches the ICMPv6 error packet rate, then the token bucket comes into play and throttling occurs, as follows.

The concept of a logical token bucket controls the rate at which ICMP messages can be transmitted. The number of tokens in the bucket is configurable, and each token represents an ICMPv6 message that can be sent. The token count is decremented each time an ICMPv6 message is sent; when the bucket reaches zero tokens, no more ICMPv6 messages can be sent until another token is added to the bucket. The default size of the token bucket is 100 tokens (packets); the range is 10 to 65535 tokens.

To change the default token bucket size or error packet rate, see the section [Configure Session Settings](#).

Control Specific ICMP or ICMPv6 Types and Codes

Use this task to create a custom ICMP or ICMPv6 application and then create a security policy rule to allow or deny that application.

STEP 1 | Create a custom application for ICMP or ICMPv6 message types and codes.

1. Select **Object > Applications** and **Add** a custom application.
2. On the **Configuration** tab, enter a **Name** for the custom application and a **Description**. For example, enter the name ping6.
3. For **Category**, select **networking**.
4. For **Subcategory**, select **ip-protocol**.
5. For **Technology**, select **network-protocol**.
6. Click **OK**.
7. On the **Advanced** tab, select **ICMP Type** or **ICMPv6 Type**.
8. For **Type**, enter the number (range is 0-255) that designates the ICMP or ICMPv6 message type you want to allow or deny. For example, Echo Request message (ping) is 128.
9. If the Type includes codes, enter the **Code** number (range is 0-255) that applies to the **Type** value you want to allow or deny. Some **Type** values have Code 0 only.
10. Click **OK**.

STEP 2 | Create a Security policy rule that allows or denies the custom application you created.

[Create a Security Policy Rule](#). On the **Application** tab, specify the name of the custom application you just created.

STEP 3 | Commit your changes.

Click **Commit**.

Configure Session Timeouts

A session timeout defines the duration of time for which PAN-OS maintains a session on the firewall after inactivity in the session. By default, when the session timeout for the protocol expires, PAN-OS closes the session. You can define a number of timeouts for TCP, UDP, and ICMP sessions in particular. The Default timeout applies to any other type of session. The timeouts are global, meaning they apply to all of the sessions of that type on the firewall.

You can also configure a global ARP cache timeout setting, which controls how long the firewall keeps ARP entries (IP address-to-hardware addresses mappings) in its cache.

In addition to the global settings, you can define timeouts for an individual application in the **Objects > Applications** tab. The firewall applies application timeouts to an application that is in established state. When configured, timeouts for an application override the global TCP or UDP session timeouts.



If you change the TCP or UDP timers at the application level, these timers for predefined applications and shared custom applications will be implemented across all virtual systems. If you need an application's timers to be different for a virtual system, you must create a custom application, assign it unique timers, and then assign the custom application to a unique virtual system.

Perform the following task if you need to change default values of the global session timeout settings for TCP, UDP, ICMP, Captive Portal authentication, or other types of sessions. All values are in seconds.



The defaults are optimal values. However, you can modify these according to your network needs. Setting a value too low could cause sensitivity to minor network delays and could result in a failure to establish connections with the firewall. Setting a value too high could delay failure detection.

STEP 1 | Access the session timeouts.

Select **Device > Setup > Session** and edit the Session Timeouts.

STEP 2 | (Optional) Change miscellaneous timeouts.

- **Default**—Maximum length of time that a non-TCP/UDP or non-ICMP session can be open without a response (range is 1 to 15,999,999; default is 30).
- **Discard Default**—Maximum length of time that a non-TCP/UDP session remains open after PAN-OS denies a session based on security policies configured on the firewall (range is 1 to 15,999,999; default is 60).
- **Scan**—Maximum length of time that any session remains open after it is considered inactive; an application is regarded as inactive when it exceeds the application trickling threshold defined for the application (range is 5 to 30; default is 10).
- **Authentication Portal**—Authentication session timeout for the Captive Portal web form. To access the requested content, the user must enter the authentication credentials in this form and be successfully authenticated (range is 1 to 15,999,999; default is 30).
- To define other Authentication Portal timeouts, such as the idle timer and the expiration time before the user must be re-authenticated, select **Device > User Identification > Authentication Portal Settings**. See [Configure Authentication Portal](#).

STEP 3 | (Optional) Change TCP timeouts.

- **Discard TCP**—Maximum length of time that a TCP session remains open after it is denied based on a security policy configured on the firewall. Range is 1 to 15,999,999; default is 90.
- **TCP**—Maximum length of time that a TCP session remains open without a response, after a TCP session is in the Established state (after the handshake is complete and/or data is being transmitted). Range is 1 to 15,999,999; default is 3,600.
- **TCP Handshake**—Maximum length of time permitted between receiving the SYN-ACK and the subsequent ACK to fully establish the session. Range is 1 to 60; default is 10.
- **TCP init**—Maximum length of time permitted between receiving the SYN and SYN-ACK prior to starting the TCP handshake timer. Range is 1 to 60; default is 5.
- **TCP Half Closed**—Maximum length of time between receiving the first FIN and receiving the second FIN or a RST. Range is 1 to 604,800; default is 120.
- **TCP Time Wait**—Maximum length of time after receiving the second FIN or a RST. Range is 1 to 600; default is 15.
- **Unverified RST**—Maximum length of time after receiving a RST that cannot be verified (the RST is within the TCP window but has an unexpected sequence number, or the RST is from an asymmetric path). Range is 1 to 600; default is 30.
- See also the **Scan** timeout in the section [\(Optional\) Change miscellaneous timeouts](#).

STEP 4 | (Optional) Change UDP timeouts.

- **Discard UDP**—Maximum length of time that a UDP session remains open after it is denied based on a security policy configured on the firewall. Range is 1 to 15,999,999; default is 60.
- **UDP**—Maximum length of time that a UDP session remains open without a UDP response. Range is 1 to 15,999,999; default is 30.
- See also the **Scan** timeout in the section [\(Optional\) Change miscellaneous timeouts](#).

STEP 5 | [\(Optional\)](#) Change ICMP timeouts.

- **ICMP**—Maximum length of time that an ICMP session can be open without an ICMP response. Range is 1 to 15,999,999; default is 6.
- See also the **Discard Default** and **Scan** timeout in the section [\(Optional\) Change miscellaneous timeouts](#).

STEP 6 | Click **OK** and **Commit**.

STEP 7 | [\(Optional\)](#) Change the ARP cache timeout.

1. Access the CLI and specify how many seconds the firewall keeps ARP entries in its cache. Use the operational command **set system setting arp-cache-timeout <value>**, where the range is 60 to 65,535; default is 1,800.

If you decrease the timeout and existing entries in the cache have a TTL greater than the new timeout, the firewall removes those entries and refreshes the ARP cache. If you increase the timeout and existing entries have a TTL less than the new timeout, they expire according to the TTL and the firewall caches new entries with the larger timeout value.

2. View the ARP cache timeout setting with the operational CLI command **show system setting arp-cache-timeout**.

Configure Session Settings

This topic describes various settings for sessions other than timeout values. Perform these tasks if you need to change the default settings.

STEP 1 | Change the session settings.

Select **Device > Setup > Session** and edit the Session Settings.

STEP 2 | Specify whether to apply newly configured Security policy rules to sessions that are in progress.

Select **Rematch all sessions on config policy change** to apply newly configured Security policy rules to sessions that are already in progress. This capability is enabled by default. If you clear this check box, any policy rule changes you make apply only to sessions initiated after you commit the policy change.

For example, if a Telnet session started while an associated policy rule was configured that allowed Telnet, and you subsequently committed a policy change to deny Telnet, the firewall applies the revised policy to the current session and blocks it.

STEP 3 | Configure IPv6 settings.

- **ICMPv6 Token Bucket Size**—Default: 100 tokens. See the section [ICMPv6 Rate Limiting](#).
- **ICMPv6 Error Packet Rate (per sec)**—Default: 100. See the section [ICMPv6 Rate Limiting](#).
- **Enable IPv6 Firewalling**—Enables firewall capabilities for IPv6. All IPv6-based configurations are ignored if IPv6 is not enabled. Even if IPv6 is enabled for an interface, the **IPv6 Firewalling** setting must also be enabled for IPv6 to function.

STEP 4 | Enable jumbo frames and set the MTU.

1. Select **Enable Jumbo Frame** to enable jumbo frame support on Ethernet interfaces. Jumbo frames have a maximum transmission unit (MTU) of 9,216 bytes and are available on certain models.
2. Set the **Global MTU**, depending on whether or not you enabled jumbo frames:
 - If you did not enable jumbo frames, the **Global MTU** defaults to 1,500 bytes; the range is 576 to 1,500 bytes.
 - If you enabled jumbo frames, the **Global MTU** defaults to 9,192 bytes; the range is 9,192 to 9,216 bytes.



Jumbo Frames can take up to five times more memory compared to normal packets and can reduce the number of available packet-buffers by 20%. This reduces the queue sizes dedicated for out of order, application identification, and other such packet processing tasks. As of PAN-OS 8.1, if you enable the jumbo frame global MTU configuration and reboot your firewall, packet buffers are then redistributed to process jumbo frames more efficiently.

If you enable jumbo frames and you have interfaces where the MTU is not specifically configured, those interfaces will automatically inherit the jumbo frame size. Therefore,

before you enable jumbo frames, if you have any interface that you do not want to have jumbo frames, you must set the MTU for that interface to 1500 bytes or another value.



If you import (**Device > Setup > Operations > Import**) and load a configuration that has Jumbo Frame enabled, and then commit to a firewall that does not already have Jumbo Frame enabled, the **Enable Jumbo Frame** setting is not committed to the configuration. You should first **Enable Jumbo Frame**, reboot, and then import, load and commit the configuration.

STEP 5 | Tune NAT session settings.

- **NAT64 IPv6 Minimum Network MTU**—Sets the global MTU for IPv6 translated traffic. The default of 1,280 bytes is based on the standard minimum MTU for IPv6 traffic.
- **NAT Oversubscription Rate**—If NAT is configured to be Dynamic IP and Port (DIPP) translation, an oversubscription rate can be configured to multiply the number of times that the same translated IP address and port pair can be used concurrently. The rate is 1, 2, 4, or 8. The default setting is based on the [firewall model](#).
 - A rate of 1 means no oversubscription; each translated IP address and port pair can be used only once at a time.
 - If the setting is **Platform Default**, user configuration of the rate is disabled and the default oversubscription rate for the model applies.

Reducing the oversubscription rate decreases the number of source device translations, but provides higher NAT rule capacities.

STEP 6 | Tune accelerated aging settings.

Select **Accelerated Aging** to enable faster aging-out of idle sessions. You can also change the threshold (%) and scaling factor:

- **Accelerated Aging Threshold**—Percentage of the session table that is full when accelerated aging begins. The default is 80%. When the session table reaches this threshold (% full), PAN-OS applies the Accelerated Aging Scaling Factor to the aging calculations for all sessions.
- **Accelerated Aging Scaling Factor**—Scaling factor used in the accelerated aging calculations. The default scaling factor is 2, meaning that the accelerated aging occurs at a rate twice as fast as the configured idle time. The configured idle time divided by 2 results in a faster timeout of one-half the time. To calculate the session's accelerated aging, PAN-OS divides the configured idle time (for that type of session) by the scaling factor to determine a shorter timeout.

For example, if the scaling factor is 10, a session that would normally time out after 3600 seconds would time out 10 times faster (in 1/10 of the time), which is 360 seconds.

STEP 7 | Enable packet buffer protection.

1. Select **Packet Buffer Protection** to enable the firewall to take action against sessions that can overwhelm the its packet buffer and causes legitimate traffic to be dropped; enabled by default.
2. If you enable packet buffer protection, you can tune the thresholds and timers that dictate how the firewall responds to packet buffer abuse.
 - **Alert (%)**: When packet buffer utilization exceeds this threshold, the firewall creates a log event. The threshold is set to 50% by default and the range is 0% to 99%. If the value is set to 0%, the firewall does not create a log event.
 - **Activate (%)**: When a packet buffer utilization exceeds this threshold, the firewall applies random early drop (RED) to abusive sessions. The threshold is set to 80% by default and the range is 0% to 99%. If the value is set to 0%, the firewall does not apply RED.



Alert events are recorded in the system log. Events for dropped traffic, discarded sessions, and blocked IP address are recorded in the threat log.

- **Block Hold Time (sec)**: The amount of time a RED-mitigated session is allowed to continue before it is discarded. By default, the block hold time is 60 seconds. The range is 0 to 65,535 seconds. If the value is set to 0, the firewall does not discard sessions based on packet buffer protection.
- **Block Duration (sec)**: This setting defines how long a session is discarded or an IP address is blocked. The default is 3,600 seconds with a range of 0 seconds to 15,999,999 seconds. If this value is set to 0, the firewall does not discard sessions or block IP addresses based on packet buffer protection.

STEP 8 | Enable buffering of multicast route setup packets.

1. Select **Multicast Route Setup Buffering** to enable the firewall to preserve the first packet in a multicast session when the multicast route or forwarding information base (FIB) entry does not yet exist for the corresponding multicast group. By default, the firewall does not buffer the first multicast packet in a new session; instead, it uses the first packet to set up the multicast route. This is expected behavior for multicast traffic. You only need to enable multicast route setup buffering if your content servers are directly connected to the firewall and your custom application cannot withstand the first packet in the session being dropped. This option is disabled by default.
2. If you enable buffering, you can also tune the **Buffer Size**, which specifies the buffer size per flow. The firewall can buffer a maximum of 5,000 packets.



*You can also tune the duration, in seconds, for which a multicast route remains in the routing table on the firewall after the session ends by configuring the multicast settings on the virtual router that handles your virtual router (set the **Multicast Route Age Out Time (sec)** on the **Multicast > Advanced** tab in the virtual router configuration).*

STEP 9 | Save the session settings.

Click **OK**.

STEP 10 | Tune the **Maximum Segment Size (MSS)** adjustment size settings for a Layer 3 interface.

1. Select **Network > Interfaces**, select **Ethernet**, **VLAN**, or **Loopback**, and select a Layer 3 interface.
2. Select **Advanced > Other Info**.
3. Select **Adjust TCP MSS** and enter a value for one or both of the following:
 - **IPv4 MSS Adjustment Size** (range is 40 to 300 bytes; default is 40 bytes).
 - **IPv6 MSS Adjustment Size** (range is 60 to 300 bytes; default is 60 bytes).
4. Click **OK**.

STEP 11 | Commit your changes.

Click **Commit**.

STEP 12 | Reboot the firewall after changing the jumbo frame configuration.

1. Select **Device > Setup > Operations**.
2. Click **Reboot Device**.

Session Distribution Policies

Session distribution policies define how PA-5200 and PA-7000 Series firewalls distribute security processing (App-ID, Content-ID, URL filtering, SSL decryption, and IPSec) among dataplane processors (DPs) on the firewall. Each policy is specifically designed for a certain type of network environment and firewall configuration to ensure that the firewall distributes sessions with maximum efficiency. For example, the Hash session distribution policy is best fit for environments that use large scale source NAT.

The number of DPs on a firewall varies based on the firewall model:

Firewall Model	Dataplane Processor(s)
PA-7000 Series	Depends on the number of installed Network Processing Cards (NPCs). Each NPC has multiple dataplane processors (DPs) and you can install multiple NPCs in the firewall.
PA-5220 firewall	1  <i>The PA-5220 firewall has only one DP so sessions distribution policies do not have an effect. Leave the policy set to the default (round-robin).</i>
PA-5250 firewall	2
PA-5260 and PA-5280 firewalls	3
PA-5450 firewall	Depends on the number of installed Data Processing Cards (DPCs).

The following topics provide information about the available session distribution policies, how to change an active policy, and how to view session distribution statistics.

- [Session Distribution Policy Descriptions](#)
- [Change the Session Distribution Policy and View Statistics](#)

Session Distribution Policy Descriptions

The following table provides information about [Session Distribution Policies](#) to help you decide which policy best fits your environment and firewall configuration.

Session Distribution Policy	Description
Fixed	Allows you to specify the dataplane processor (DP) that the firewall will use for security processing. Use this policy for debugging purposes.

Session Distribution Policy	Description
Hash	<p>The firewall distributes sessions based on a hash of the source address or destination address. Hash based distribution improves the efficiency of NAT address resource management and reduces latency for NAT session setup by avoiding potential IP address or port conflicts.</p> <p>Use this policy in environments that use large scale source NAT with dynamic IP translation or Dynamic IP and Port translation or both. When using dynamic IP translation, select the source address option. When using dynamic IP and port translation, select the destination address option.</p>
Ingress-slot (default on PA-7000 Series firewalls)	<p>(PA-7000 Series firewalls only) New sessions are assigned to a DP on the same NPC on which the first packet of the session arrived. The selection of the DP is based on the session-load algorithm but, in this case, sessions are limited to the DPs on the ingress NPC.</p> <p>Depending on the traffic and network topology, this policy generally decreases the odds that traffic will need to traverse the switch fabric.</p> <p>Use this policy to reduce latency if both ingress and egress are on the same NPC. If the firewall has a mix of NPCs (PA-7000 20G and PA-7000 20GXM for example), this policy can isolate the increased capacity to the corresponding NPCs and help to isolate the impact of NPC failures.</p>
Random	The firewall randomly selects a DP for session processing.
Round-robin (default on PA-5200 Series firewalls)	<p>The firewall selects the dataplane processor based on a round-robin algorithm between active dataplanes so that input, output, and security processing functions are shared among all dataplanes.</p> <p>Use this policy in low to medium demand environments where a simple and predictable load balancing algorithm will suffice.</p> <p>In high demand environments, we recommend that you use the session-load algorithm.</p>
Session-load	This policy is similar to the round-robin policy but uses a weight-based algorithm to determine how to distribute sessions to achieve balance among the DPs. Because of the variability in the lifetime of a session, the DPs may not always experience an equal load. For example, if the

Session Distribution Policy	Description
	<p>firewall has three DPs and DP0 is at 25% of capacity, DP1 is at 25%, and DP2 is at 50%, new session assignment will be weighted towards the DP with the lower capacities. This helps improve load balancing over time.</p> <p>Use this policy in environments where sessions are distributed across multiple NPC slots, such as in an inter-slot aggregate interface group or environments with asymmetric forwarding. You can also use this policy or the ingress-slot policy if the firewall has a combination of NPCs with different session capacities (such as a combination of PA-7000 20G and PA-7000 20GXM NPCs).</p>
Symmetric-hash	<p>(PA-5200 Series and PA-7000 Series firewalls running PAN-OS 8.0 or later) The firewall selects the DP by a hash of sorted source and destination IP addresses. This policy provides the same results for server-to-client (s2c) and client-to-server (c2s) traffic (assuming the firewall does not use NAT).</p> <p>Use this policy in high-demand IPSec or GTP deployments.</p> <p>With these protocols, each direction is treated as a unidirectional flow where the flow tuples cannot be derived from each other. This policy improves performance and reduces latency by ensuring that both directions are assigned to the same DP, which removes the need for inter-DP communication.</p>

Change the Session Distribution Policy and View Statistics

The following table describes how to view and change the active [Session Distribution Policies](#) and describes how to view session statistics for each dataplane processor (DP) in the firewall.

Task	Command
Show the active session distribution policy.	<p>Use the show session distribution policy command to view the active session distribution policy.</p> <p>The following output is from a PA-7080 firewall with four NPCs installed in slots 2, 10, 11, and 12 with the ingress-slot distribution policy enabled:</p> <pre data-bbox="572 1790 1176 1826">> show session distribution policy</pre>

Task	Command
	<p>Ownership Distribution Policy: ingress-slot</p> <p>Flow Enabled Line Cards: [2, 10, 11, 12] Packet Processing Enabled Line Cards: [2, 10, 11, 12]</p>
Change the active session distribution policy.	<p>Use the set session distribution-policy <policy> command to change the active session distribution policy.</p> <p>For example, to select the session-load policy, enter the following command:</p> <pre>> set session distribution-policy session-load</pre>
View session distribution statistics.	<p>Use the show session distribution statistics command to view the dataplane processors (DPs) on the firewall and the number of sessions on each active DP.</p> <p>The following output is from a PA-7080 firewall:</p> <pre>> show session distribution statistics DP Active Dispatched Dispatched/sec ----- s1dp0 78698 7829818 1473 s1dp1 78775 7831384 1535 s3dp0 7796 736639 1488 s3dp1 7707 737026 1442</pre> <p>The DP Active column lists each dataplane on the installed NPCs. The first two characters indicate the slot number and the last three characters indicate the dataplane number. For example, s1dp0 indicates dataplane 0 on the NPC in slot 1 and s1dp1 indicates dataplane 1 on the NPC in slot1.</p> <p>The Dispatched column shows the total number of sessions that the dataplane processed since the last time the firewall rebooted.</p> <p>The Dispatched/sec column indicates the dispatch rate. If you add the numbers in the Dispatched column, the total equals the number of active sessions on the firewall. You can also view the total number of active sessions by running the show session info CLI command.</p> <p> The PA-5200 Series firewall output will look similar, except that the number of DPs depends on the model and there is only one NPC slot (s1).</p>

Prevent TCP Split Handshake Session Establishment

You can configure a [TCP Split Handshake Drop](#) in a Zone Protection profile to prevent TCP sessions from being established unless they use the standard three-way handshake. This task assumes that you assigned a security zone for the interface where you want to prevent TCP split handshakes from establishing a session.

STEP 1 | Configure a Zone Protection profile to prevent TCP sessions that use anything other than a three-way handshake to establish a session.

1. Select **Network > Network Profiles > Zone Protection** and **Add** a new profile (or select an existing profile).
2. If creating a new profile, enter a **Name** for the profile and an optional **Description**.
3. Select **Packet Based Attack Protection > TCP Drop** and select **Split Handshake**.
4. Click **OK**.

STEP 2 | Apply the profile to one or more security zones.

1. Select **Network > Zones** and select the zone where you want to assign the zone protection profile.
2. In the Zone window, from the **Zone Protection Profile** list, select the profile you configured in the previous step.

Alternatively, you could start creating a new profile here by clicking **Zone Protection Profile**, in which case you would continue accordingly.

3. Click **OK**.
4. **(Optional)** Repeat steps 1-3 to apply the profile to additional zones.

STEP 3 | Commit your changes.

Click **OK** and **Commit**.

Tunnel Content Inspection

The firewall can inspect the traffic content of cleartext tunnel protocols without terminating the tunnel:

- Generic Routing Encapsulation (GRE) ([RFC 2784](#))
- Non-encrypted IPSec traffic [[NULL Encryption Algorithm for IPSec \(RFC 2410\)](#) and transport mode AH IPSec]
- General Packet Radio Service (GPRS) Tunneling Protocol for User Data ([GTP-U](#))
- Virtual Extensible Local Area Network (VXLAN) ([RFC 7348](#))



Tunnel content inspection is for cleartext tunnels, not for VPN or LVPN tunnels, which carry encrypted traffic.

You can use tunnel content inspection to enforce Security, DoS Protection, and QoS policies on traffic in these types of tunnels and traffic nested within another cleartext tunnel (for example, a Null Encrypted IPSec tunnel inside a GRE tunnel). You can view tunnel inspection logs and tunnel activity in the ACC to verify that tunneled traffic complies with your corporate security and usage policies.

All firewall models support tunnel content inspection for GRE, non-encrypted IPSec, and VXLAN protocols. Only [firewalls that support GTP security](#) support GTP-U tunnel content inspection—see the PAN-OS Releases by Model that Support GTP and SCTP Security in the [Compatibility Matrix](#).

By default, supported firewalls perform tunnel acceleration to improve performance and throughput for traffic going through GRE tunnels, VXLAN tunnels, and GTP-U tunnels. Tunnel acceleration provides hardware offloading to reduce the time it takes to perform flow lookups and allows the tunnel traffic to be distributed more efficiently based on the inner traffic. However, you can [Disable Tunnel Acceleration](#) to troubleshoot.

- [Tunnel Content Inspection Overview](#)
- [Configure Tunnel Content Inspection](#)
- [View Inspected Tunnel Activity](#)
- [View Tunnel Information in Logs](#)
- [Create a Custom Report Based on Tagged Tunnel Traffic](#)
- [Tunnel Acceleration Behavior](#)
- [Disable Tunnel Acceleration](#)

Tunnel Content Inspection Overview

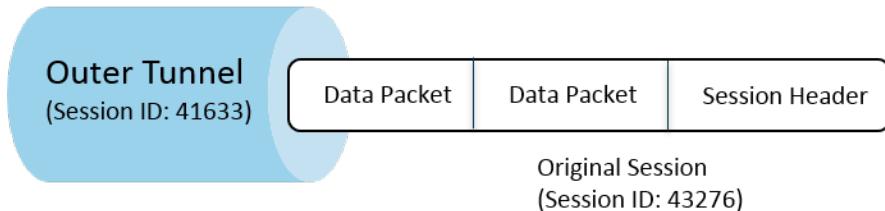
Your firewall can inspect tunnel content anywhere on the network where you do not have the opportunity to terminate the tunnel first. As long as the firewall is in the path of a GRE, non-encrypted IPSec, GTP-U, or [VXLAN](#) tunnel, the firewall can inspect the tunnel content.

- Enterprise customers who want tunnel content inspection can have some or all of the traffic on the firewall tunneled using GRE, VXLAN, or non-encrypted IPSec. For security, QoS, and reporting reasons, you want to inspect the traffic inside the tunnel.
- Service Provider customers use GTP-U to tunnel data traffic from mobile devices. You want to inspect the inner content without terminating the tunnel protocol, and you want to record user data from your users.

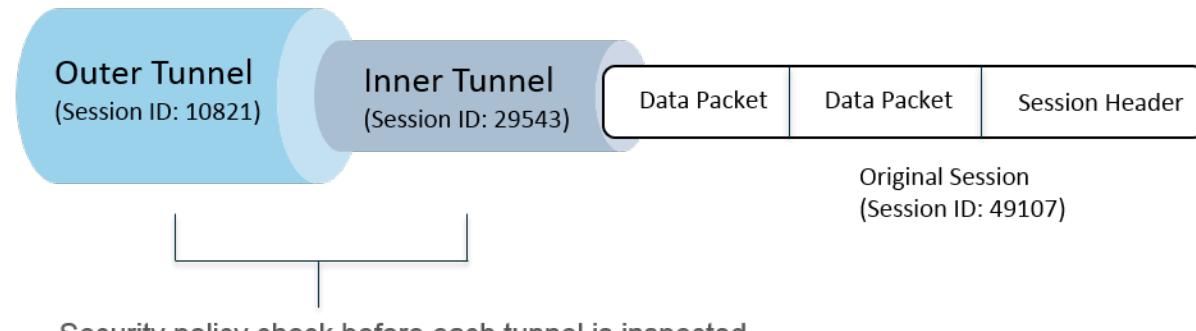
The firewall supports tunnel content inspection on Ethernet interfaces, subinterfaces, AE interfaces, VLAN interfaces, and VPN and LVPN tunnel interfaces. (The cleartext tunnel that the firewall inspects can be inside a VPN or LVPN tunnel that terminates at the firewall, hence a VPN or LVPN tunnel interface. In other words, when the firewall is a VPN or LVPN endpoint, the firewall can inspect the traffic of any non-encrypted tunnel protocols that tunnel content inspection supports.)

Tunnel content inspection is supported in Layer 3, Layer 2, virtual wire, and tap deployments. Tunnel content inspection works on shared gateways and on virtual system-to-virtual system communications.

Single Tunnel



Tunnel-in-Tunnel



The preceding figure illustrates the two levels of tunnel inspection the firewall can perform. When a firewall configured with Tunnel Inspection policy rules receives a packet:

- The firewall first performs a Security policy check to determine whether the tunnel protocol (Application) in the packet is permitted or denied. (IPv4 and IPv6 packets are supported protocols inside the tunnel.)
- If the Security policy allows the packet, the firewall matches the packet to a Tunnel Inspection policy rule based on source zone, source address, source user, destination zone, and destination address. The Tunnel Inspection policy rule determines the tunnel protocols that the firewall inspects, the maximum level of encapsulation allowed (a single tunnel or a tunnel within a tunnel), whether to allow packets containing a tunnel protocol that doesn't pass strict header inspection per [RFC 2780](#), and whether to allow packets containing unknown protocols.
- If the packet passes the Tunnel Inspection policy rule's match criteria, the firewall inspects the inner content, which is subject to your Security policy (**required**) and optional policies you can specify. (The supported policy types for the original session are listed in the following table).
- If the firewall instead finds another tunnel, the firewall recursively parses the packet for the second header and is now at level two of encapsulation, so the second tunnel inspection policy rule, which matches a tunnel zone, must allow a maximum tunnel inspection level of two levels for the firewall to continue processing the packet.
 - If your rule allows two levels of inspection, the firewall performs a Security policy check on this inner tunnel and then the Tunnel Inspection policy check. The tunnel protocol you use in an inner tunnel can differ from the tunnel protocol you use in the outer tunnel.
 - If your rule doesn't allow two levels of inspection, the firewall bases its action on whether you configured it to drop packets that have more levels of encapsulation than the maximum tunnel inspection level you configured.

By default, the content encapsulated in a tunnel belongs to the same security zone as the tunnel, and is subject to the Security policy rules that protect that zone. However, you can configure a *tunnel zone*, which gives you the flexibility to configure Security policy rules for inside content that differ from the Security policy rules for the tunnel. If you use a different tunnel inspection policy for the tunnel zone, it must always have a maximum tunnel inspection level of two levels because by definition the firewall is looking at the second level of encapsulation.

The firewall doesn't support a Tunnel Inspection policy rule that matches traffic for a tunnel that terminates on the firewall; the firewall discards packets that match the inner tunnel session. For example, when an IPSec tunnel terminates on the firewall, don't create a Tunnel Inspection policy rule that matches the tunnel you terminate. The firewall already inspects the inner tunnel traffic so no Tunnel Inspection policy rule is needed.



Although tunnel content inspection works on shared gateways and on virtual system-to-virtual system communications, you can't assign tunnel zones to shared gateways or virtual system-to-virtual system communications; they are subject to the same Security policy rules as the zones to which they belong.

Both the inner tunnel sessions and the outer tunnel sessions count toward the maximum session capacity for the firewall model.

The following table indicates with a check mark which types of policy you can apply to an outer tunnel session, an inner tunnel session, and the inside, original session:

Policy Type	Outer Tunnel Session	Inner Tunnel Session	Inside, Original Session
App-Override	✓ VXLAN Only	—	✓
DoS Protection	✓	✓	✓
NAT	✓	—	—
Policy-Based Forwarding (PBF) and Symmetric Return	✓	—	—
QoS	—	—	✓
Security (required)	✓	✓	✓
User-ID	✓	✓	✓
Zone Protection	✓	✓	✓

VXLAN is different than other protocols. The firewall can use either of two different sets of session keys to create outer tunnel sessions for VXLAN.

- VXLAN UDP Session—A six-tuple key (zone, source IP, destination IP, protocol, source port, and destination port) creates a VXLAN UDP Session.
- VNI Session—A five-tuple key that incorporates the tunnel ID (the VXLAN Network Identifier, or VNI) and uses zone, source IP, destination IP, protocol, and tunnel ID (VNI) to create a VNI Session.

You can [View Inspected Tunnel Activity](#) on the ACC or [View Tunnel Information in Logs](#). To facilitate quick viewing, configure a Monitor tag so you can monitor tunnel activity and filter log results by that tag.

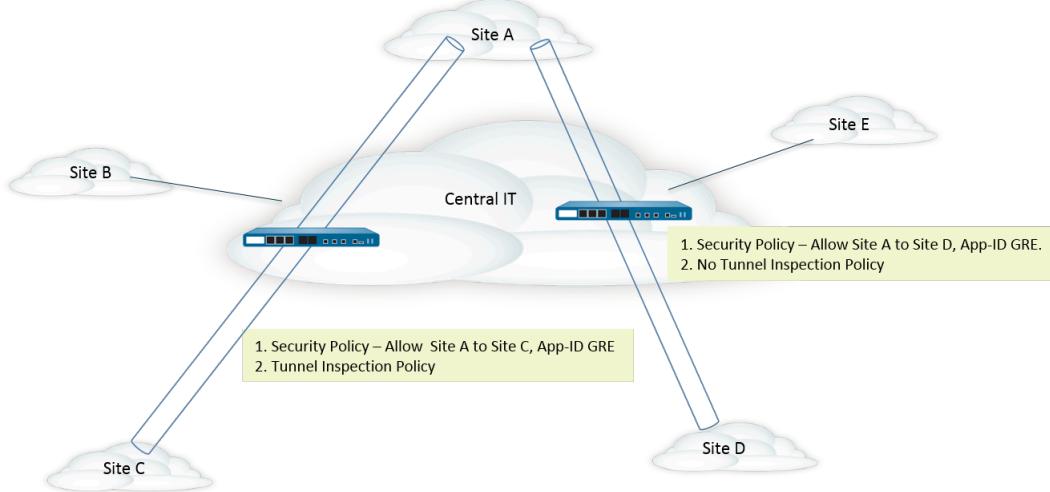
The ACC tunnel activity provides data in various views. For the Tunnel ID Usage, Tunnel Monitor Tag, and Tunnel Application Usage, the data for **bytes**, **sessions**, **threats**, **content**, and **URLs** come from the Traffic Summary database. For the Tunnel User, Tunneled Source IP and Tunneled Destination IP Activity, data for **bytes** and **sessions** come from Traffic Summary database, data for **threats** come from the Threat Summary, data for **URLs** come from the URL Summary, and data for **contents** come from the Data database, which is a subset of the Threat logs.

If you enable NetFlow on the interface, NetFlow will capture statistics for the outer tunnel only, to avoid double-counting (counting bytes of both outer and inner flows).

For the Tunnel Inspection policy rule and tunnel zone capacities for your firewall model, see the [Product Selection tool](#).

Tunnel Content Inspection

The following figure illustrates a corporation that runs multiple divisions and uses different Security policies and a Tunnel Inspection policy. A Central IT team provides connectivity between regions. A tunnel connects Site A to Site C; another tunnel connects Site A to Site D. Central IT places a firewall in the path of each tunnel; the firewall in the tunnel between Sites A and C performs tunnel inspection; the firewall in the tunnel between Sites A and D has no tunnel inspection policy because the traffic is very sensitive.



Configure Tunnel Content Inspection

Perform this task to configure tunnel content inspection for a tunnel protocol that you allow through a tunnel.

STEP 1 | Create a Security policy rule to allow packets that use a specific application (such as the GRE application) through the tunnel from the source zone to the destination zone.

Create a Security policy rule

 *The firewall can create tunnel inspection logs at the start of a session, at the end of a session, or both. When you specify **Actions** for the Security policy rule, select **Log at Session Start** for long-lived tunnel sessions, such as GRE sessions.*

STEP 2 | Create a tunnel inspection policy rule.

1. Select **Policies > Tunnel Inspection** and **Add** a policy rule.
2. On the **General** tab, enter a tunnel inspection policy rule **Name**, beginning with an alphanumeric character and containing zero or more alphanumeric, underscore, hyphen, period, and space characters.
3. **(Optional)** Enter a **Description**.
4. **(Optional)** For reporting and logging purposes, specify a **Tag** that identifies the packets that are subject to the Tunnel Inspection policy rule.

STEP 3 | Specify the criteria that determine the source of packets to which the tunnel inspection policy rule applies.

1. Select the **Source** tab.
2. **Add a Source Zone** from the list of zones (default is **Any**).
3. **(Optional) Add a Source Address**. You can enter an IPv4 or IPv6 address, an address group, or a Geo Region address object (**Any**).
4. **(Optional) Select Negate** to choose any addresses except those you specify.
5. **(Optional) Add a Source User** (default is **any**). **Known-user** is a user who has authenticated; an **Unknown** user has not authenticated.

STEP 4 | Specify the criteria that determine the destination of packets to which the tunnel inspection policy rule applies.

1. Select the **Destination** tab.
2. **Add a Destination Zone** from the list of zones (default is **Any**).
3. **(Optional) Add a Destination Address**. You can enter an IPv4 or IPv6 address, an address group, or a Geo Region address object (default is **Any**).
You can also configure a new address or address group.
4. **(Optional) Select Negate** to choose any addresses except those you specify.

STEP 5 | Specify the tunnel protocols that the firewall will inspect for this rule.

1. Select the **Inspection** tab.
2. Add one or more tunnel **Protocols** that you want the firewall to inspect:
 - **GRE**—Firewall inspects packets that use Generic Route Encapsulation (GRE) in the tunnel.
 - **GTP-U**—Firewall inspects packets that use General Packet Radio Service (GPRS) Tunneling Protocol for User Data (GTP-U) in the tunnel.
 - **Non-encrypted IPSec**—Firewall inspects packets that use non-encrypted IPSec (Null Encrypted IPSec or transport mode AH IPSec) in the tunnel.
 - **VXLAN**—Firewall inspects packets that use the Virtual Extensible Local Area Network (VXLAN) tunneling protocol in the tunnel.

STEP 6 | Specify how many levels of encapsulation the firewall inspects and the conditions under which the firewall drops a packet.

1. Select **Inspect Options**.
2. Select the **Maximum Tunnel Inspection Levels** that the firewall will inspect:
 - **One Level** (default)—Firewall inspects content that is in the outer tunnel only.
For VXLAN, the firewall inspects a VXLAN payload to find the encapsulated content or applications within the tunnel. You must select **One Level** because VXLAN inspection only occurs on the outer tunnel.
 - **Two Levels (Tunnel In Tunnel)**—Firewall inspects content that is in the outer tunnel and content that is in the inner tunnel.
3. Select any, all, or none of the following to specify whether the firewall drops a packet under each condition:
 - **Drop packet if over maximum tunnel inspection level**—Firewall drops a packet that contains more levels of encapsulation than are configured for **Maximum Tunnel Inspection Levels**.
 - **Drop packet if tunnel protocol fails strict header check**—Firewall drops a packet that contains a tunnel protocol that uses a header that is non-compliant with the RFC for the protocol. Non-compliant headers can indicate suspicious packets. This option causes the firewall to verify GRE headers against RFC 2890.
 If your firewall is tunneling GRE with a device that implements a version of GRE older than [RFC 2890](#), you should not enable the option to **Drop packet if tunnel protocol fails strict header check**.
 - **Drop packet if unknown protocol inside tunnel**—Firewall drops a packet that contains a protocol inside the tunnel that the firewall can't identify.
For example, if this option is selected, the firewall drops encrypted IPSec packets that match the tunnel inspection policy rule because the firewall can't read them. Thus, you can allow IPSec packets and the firewall will allow only null-encrypted IPSec and AH IPSec packets.
 - **Return scanned VXLAN tunnel to source**—When traffic is redirected (steered) to the firewall, VXLAN encapsulates the packet. Traffic steering is most common in public

cloud environments. Enable **Return scanned VXLAN tunnel to source** to return the encapsulated packet to the originating VXLAN tunnel endpoint (VTEP). This option is only supported on Layer 3, Layer 3 subinterface, aggregate interface Layer 3, and VLAN.

4. Click **OK**.

STEP 7 | Manage tunnel inspection policy rules.

Use the following to manage tunnel inspection policy rules:

- (**Filter field**)—Displays only the tunnel policy rules named in the filter field.
- **Delete**—Removes selected tunnel policy rules.
- **Clone**—An alternative to the **Add** button; duplicates the selected rule with a new name, which you can then revise.
- **Enable**—Enables the selected tunnel policy rules.
- **Disable**—Disables the selected tunnel policy rules.
- **Move**—Moves the selected tunnel policy rules up or down in the list; packets are evaluated against the rules in order from the top down.
- **Highlight Unused Rules**—Highlights tunnel policy rules that no packets have matched since the last time the firewall was restarted.

STEP 8 | **(Optional)** Create a tunnel source zone and tunnel destination zone for tunnel content and configure a Security policy rule for each zone.

 *The best practice is to create tunnel zones for your tunnel traffic. Thus, the firewall creates separate sessions for tunneled and non-tunneled packets that have the same five-tuple (source IP address and port, destination IP address and port, and protocol).*

 *Assigning tunnel zones to tunnel traffic on a PA-5200 Series firewall causes the firewall to do tunnel inspection in software; tunnel inspection is not offloaded to hardware.*

1. If you want tunnel content to be subject to Security policy rules that are different from the Security policy rules for the zone of the outer tunnel (configured earlier), select **Network > Zones** and **Add a Name** for the Tunnel Source Zone.
2. For **Location**, select the virtual system.
3. For **Type**, select **Tunnel**.
4. Click **OK**.
5. Repeat these substeps to create the Tunnel Destination Zone.
6. [Configure a Security policy rule](#) for the Tunnel Source Zone.

 *Because you might not know the originator of the tunnel traffic or the direction of the traffic flow and you don't want to inadvertently prohibit traffic for an application through the tunnel, specify both tunnel zones as the **Source Zone** and both tunnel zones as the **Destination Zone** in your Security policy rule, or select **Any** for both the source and destination zones; then specify the **Applications**.*

7. [Configure a Security policy rule](#) for the Tunnel Destination Zone. The tip in the previous step for configuring a Security policy rule for the Tunnel Source Zone applies to the Tunnel Destination Zone, as well.

STEP 9 | **(Optional)** Specify the Tunnel Source Zone and Tunnel Destination Zone for the inner content.

1. Specify the Tunnel Source Zone and Tunnel Destination Zone (that you just added) for the inner content. Select **Policies > Tunnel Inspection** and on the **General** tab, select the **Name** of the tunnel inspection policy rule you created.
2. Select **Inspection**.
3. Select **Security Options**.
4. **Enable Security Options** (disabled by default) to cause the inner content source to belong to the **Tunnel Source Zone** you specify and to cause the inner content destination to belong to the **Tunnel Destination Zone** you specify.

If you don't **Enable Security Options**, the inner content source belongs to the same source zone as the outer tunnel source and the inner content destination belongs to the same destination zone as the outer tunnel destination, which means they are subject to the same Security policy rules that apply to those outer zones.

5. For **Tunnel Source Zone**, select the appropriate tunnel zone you created in the previous step so that the policies associated with that zone apply to the tunnel source zone.

Otherwise, by default, the inner content will use the same source zone that is used in the outer tunnel and the policies of the outer tunnel source zone apply to the inner content source zone, as well.

6. For **Tunnel Destination Zone**, select the appropriate tunnel zone you created in the previous step so that the policies associated with that zone apply to the tunnel destination zone. Otherwise, by default, the inner content will use the same destination zone that is used in the outer tunnel and the policies of the outer tunnel destination zone apply to the inner content destination zone, as well.



If you configure a **Tunnel Source Zone** and **Tunnel Destination Zone** for the tunnel inspection policy rule, you should configure a specific **Source Zone** (in Step 3) and a specific **Destination Zone** (in Step 4) in the match criteria of the tunnel inspection policy rule, instead of specifying a **Source Zone** of Any and a **Destination Zone** of Any. This tip ensures the direction of zone reassignment corresponds appropriately to the parent zones.



On a PA-5200 Series or PA-7080 firewall, if you use multicast underlay while inspecting VXLAN, the inner session would be duplicated on multiple dataplanes and a race condition could happen. To avoid the drop of some packets, the following requirements apply:

- You must configure a separate tunnel content inspection rule to match outer VXLAN packets going to each VXLAN tunnel endpoint (VTEP).
- In the separate rule, you assign a tunnel zone. Using a different tunnel zone would make the inner session different for each endpoint. The race condition would not happen, and no packet drop would be seen.

7. Click **OK**.

STEP 10 | Set monitoring options for traffic that matches a tunnel inspection policy rule.

1. Select **Policies > Tunnel Inspection** and select the tunnel inspection policy rule you created.
2. Select **Inspection > Monitor Options**.
3. Enter a **Monitor Name** to group similar traffic together for purposes of logging and reporting.
4. Enter a **Monitor Tag (number)** to group similar traffic together for logging and reporting (range is 1 to 16,777,215). The tag number is globally defined.



This field does not apply to the VXLAN protocol. VXLAN logs automatically use the VNI ID from the VXLAN header.



If you tag tunnel traffic, you can later filter on the Monitor Tag in the tunnel inspection log and use the ACC to view tunnel activity based on Monitor Tag.

5. **Override Security Rule Log Setting** to enable logging and log forwarding options for sessions that meet the selected tunnel inspection policy rule. If you don't select this setting, tunnel log generation and log forwarding are determined by the log settings for the Security policy rule that applies to the tunnel traffic. You can override log forwarding settings in Security policy rules that control traffic logs by configuring tunnel inspection log settings to store tunnel logs separately from traffic logs. The tunnel inspection logs

store the outer tunnel (GRE, non-encrypted IPSec, VXLAN, or GTP-U) sessions and the traffic logs store the inner traffic flows.

6. Select **Log at Session Start** to log traffic at the start of a session.



The best practice for Tunnel logs is to log both at session start and session end because tunnels can stay up for long periods of time. For example, GRE tunnels can come up when the router boots and never terminate until the router is rebooted. If you don't log at session start, you will never see in the ACC that there is an active GRE tunnel.

7. Select **Log at Session End** to log traffic at the end of a session.
8. Select a **Log Forwarding** profile that determines where the firewall forwards tunnel logs for sessions that meet the tunnel inspection rule. Alternatively, you can create a new Log Forwarding profile if you [Configure Log Forwarding](#).
9. Click **OK**.

STEP 11 | (Optional, VXLAN Only) Configure a VXLAN ID (VNI). By default, all VXLAN network interfaces (VNIs) are inspected. If you configure one or more VXLAN IDs, the policy inspects only those VNIs.



Only the VXLAN protocol uses the Tunnel ID tab to specify the VNI.

1. Select the **Tunnel ID** tab and click **Add**.
2. Assign a **Name**. The name is a convenience, and is not a factor in logging, monitoring, or reporting.
3. In the **VXLAN ID (VNI)** field, enter a single VNI, a comma-separated list of VNIs, a range of VNIs (with a hyphen as the separator), or a combination of these. For example, you can specify:

1677002,1677003,1677011-1677038,1024

STEP 12 | (Optional) If you enabled **Rematch Sessions** (**Device > Setup > Session**), ensure the firewall doesn't drop existing sessions when you create or revise a tunnel inspection policy by disabling **Reject Non-SYN TCP** for the zones that control your tunnel Security policy rules.

The firewall displays the following warning when you:

- Create a tunnel inspection policy rule.
- Edit a tunnel inspection policy rule by adding a **Protocol** or by increasing the **Maximum Tunnel Inspection Levels** from **One Level** to **Two Levels**.
- **Enable Security Options** in the **Security Options** tab by either adding new zones or changing one zone to another zone.

— Warning: Enabling tunnel inspection policies on existing tunnel sessions will cause existing TCP sessions inside the tunnel to be treated as non-syn-tcp flows. To ensure existing sessions are not dropped when the tunnel inspection policy is enabled, set the **Reject Non-SYN TCP** setting for the zone(s) to **no** using a Zone Protection profile and apply it to the zones that control the tunnel's security policies. Once the existing sessions have been recognized by the firewall, you can re-enable the **Reject Non-SYN TCP** setting by setting it to **yes** or **global**.

1. Select **Network > Network Profiles > Zone Protection** and **Add** a profile.
2. Enter a **Name** for the profile.
3. Select **Packet Based Attack Protection > TCP Drop**.
4. For **Reject Non-SYN TCP**, select **no**.
5. Click **OK**.
6. Select **Network > Zones** and select the zone that controls your tunnel Security policy rules.
7. For **Zone Protection Profile**, select the Zone Protection profile you just created.
8. Click **OK**.
9. Repeat the previous three substeps (12.f, 12.g, and 12.h) to apply the Zone Protection profile to additional zones that control your tunnel Security policy rules.
10. After the firewall has recognized the existing sessions, you can re-enable **Reject Non-SYN TCP** by setting it to **yes** or **global**.

STEP 13 | (Optional) Limit fragmentation of traffic in a tunnel.

1. Select **Network > Network Profiles > Zone Protection** and **Add** a profile by **Name**.
2. Enter a **Description**.
3. Select **Packet Based Attack Protection > IP Drop > Fragmented traffic**.
4. Click **OK**.
5. Select **Network > Zones** and select the tunnel zone where you want to limit fragmentation.
6. For **Zone Protection Profile**, select the profile you just created to apply the Zone Protection profile to the tunnel zone.
7. Click **OK**.

STEP 14 | Commit your changes.

View Inspected Tunnel Activity

Perform the following task to view activity of inspected tunnels.

STEP 1 | Select **ACC** and select a **Virtual System** or **All** virtual systems.

STEP 2 | Select Tunnel Activity.

STEP 3 | Select a Time period to view, such as Last 24 Hrs or Last 30 Days.

STEP 4 | For Global Filters, click the + or - buttons to use ACC Filters on tunnel activity.

STEP 5 | View inspected tunnel activity; you can display and sort data in each window by **bytes**, **sessions**, **threats**, **content**, or **URLs**. Each window displays a different aspect of tunnel data in graph and table format:

- **Tunnel ID Usage**—Each tunnel protocol lists the Tunnel IDs of tunnels using that protocol. Tables provide totals of Bytes, Sessions, Threats, Content, and URLs for the protocol. Hover over the tunnel ID to get a breakdown per tunnel ID.
- **Tunnel Monitor Tag**—Each tunnel protocol lists tunnel monitor tags of tunnels using that tag. Tables provide totals of Bytes, Sessions, Threats, Content, and URLs for the tag and for the protocol. Hover over the tunnel monitor tag to get a breakdown per tag.
- **Tunneled Application Usage**—Application categories graphically display types of applications grouped into media, general interest, collaboration, and networking, and color-coded by their risk. The Application tables also include a count of users per application.
- **Tunneled User Activity**—Displays a graph of bytes sent and bytes received, for example, along an x-axis of date and time. Hover over a point on the graph to view data at that point. The Source User and Destination User table provides data per user.
- **Tunneled Source IP Activity**—Displays graphs and tables of bytes, sessions, and threats, for example, from an Attacker at an IP address. Hover over a point on the graph to view data at that point.
- **Tunneled Destination IP Activity**—Displays graphs and tables based on destination IP addresses. View threats per Victim at an IP address, for example. Hover over a point on the graph to view data at that point.

View Tunnel Information in Logs

You can view Tunnel Inspection logs themselves or view tunnel inspection information in other types of logs.

GRE, Non-Encrypted IPSec, and GTP-U Protocols

- When there is a TCI traffic rule match, GRE, IPSec, and GTP-U protocols are logged in the Tunnel Inspection log with the Tunnel log type, the matched protocol, and the configured Monitor name and Monitor tag (number).
- When there is no TCI rule match, all protocols are logged under Traffic logs.

VXLAN Protocol

- When there is a TCI traffic rule match, VXLAN protocol is logged in the Tunnel Inspection log with the Tunnel (VXLAN) log type, the configured Monitor name, and the Tunnel ID (VNI).

In the Traffic log for the inner session, the Tunnel Inspected flag indicates a VNI session. The Parent Session is the session that was active when the inner session was created so the ID might not match the current Session ID.

- When there is no TCI rule match, VNI sessions are logged in Traffic logs with the UDP protocol, source port 0, and destination port 4789 (the default).
- View Tunnel inspection logs.
 1. Select **Monitor > Logs > Tunnel Inspection** and view the log data to identify the tunnel **Applications** used in your traffic and any concerns, such as high counts for packets failing Strict Checking of headers.
 2. Click the Detailed Log View () to see details about a log.
- View other logs for tunnel inspection information.
 1. Select **Monitor > Logs**.
 2. Select **Traffic, Threat, URL Filtering, WildFire Submissions, Data Filtering, or Unified**.
 3. For a log entry, click the Detailed Log View () .
 4. In the Flags window, see if the **Tunnel Inspected** flag is checked. A Tunnel Inspected flag indicates the firewall used a Tunnel Inspection policy rule to inspect the inside content or inner tunnel. Parent Session information refers to an outer tunnel (relative to an inner tunnel) or an inner tunnel (relative to inside content).

On the **Traffic, Threat, URL Filtering, WildFire Submissions, Data Filtering** logs, only direct parent information appears in the Detailed Log View of the inner session log, no tunnel log information. If you configured two levels of tunnel inspection, you can select the parent session of this direct parent to view the second parent log. (You must monitor the **Tunnel Inspection** log as shown in the prior step to view tunnel log information.)

Create a Custom Report Based on Tagged Tunnel Traffic

You can create a report to gather information based on the tag you applied to tunnel traffic.

STEP 1 | Select **Monitor > Manage Custom Reports** and click **Add**.

STEP 2 | For Database, select the Traffic, Threat, URL, Data Filtering, or WildFire Submissions log.

STEP 3 | For Available Columns, select Flags and Monitor Tag, along with other data you want in the report.

You can also [Generate Custom Reports](#).

Tunnel Acceleration Behavior

The following sections provide background information about GTP-U, GRE, and VXLAN tunnel acceleration, which may be helpful to know before you decide to [Disable Tunnel Acceleration](#).

- [GTP-U](#)
- [GRE](#)
- [VXLAN](#)

GTP-U

Criteria that must be met before GTP tunnel acceleration is enabled:

1. Generic tunnel acceleration is enabled under **Device > Setup > Management** (in General Settings, Tunnel Acceleration is checked).
2. GTP Security is enabled under **Device > Setup > Management** (in General Settings, GTP Security is checked).
3. No Tunnel Inspection policy rule with GTP-U protocol is enabled.
4. After you commit the configuration, you must reboot to load the GTP-U parser program.

Criteria for identifying GTP-U packets in hardware:

1. UDP destination port is 2152.
2. GTP.version is 1 and GTP.protocol_type is 1.

How tunnel acceleration alters the flow ID:

- If GTP-U packet passes both identification criteria, the firewall sets the following in flow key:
 - Encoding bit: 1
 - UDP destination port: tunnel endpoint identifier (TEID)
 - Source address: 0
- Otherwise, the packet is processed as a normal UDP packet.

Benefits of GTP-U Tunnel Acceleration

If GTP-U acceleration is enabled, the main benefit occurs if there is a lot of tunneled traffic that can be offloaded. A large percentage of GTP traffic is sourced from mobile devices and is mostly web traffic, which won't be offloaded when the inner payload is inspected.

The GTP Security feature is fully functional without acceleration and the performance benefit is tied to the amount of inner payload traffic that can be offloaded by the hardware. For example, anything that would normally get marked as L7 complete will be offloaded and handled solely in hardware as an inner application inside of GTP.

GRE

Criterion for tunnel acceleration taking effect with GRE:

- Generic tunnel acceleration is enabled under **Device > Setup > Management** (in General Settings, Tunnel Acceleration is checked).

Criterion for identifying GRE packets in hardware:

- IP protocol 47

How tunnel acceleration alters the flow ID:

- Flow key is the same with and without tunnel acceleration.

Benefits of GRE Tunnel Acceleration

- **With TCI:** GRE passthrough traffic will see approximately 30% increase in performance in flow handling with tunnel acceleration compared to the same traffic without tunnel acceleration.
- **Without TCI:** There is no performance impact for GRE traffic when disabling tunnel acceleration if no tunnel content inspection (TCI) policies are being used.

VXLAN

Criterion for tunnel acceleration taking effect with VXLAN:

- Generic tunnel acceleration is enabled under **Device > Setup > Management** (in General Settings, Tunnel Acceleration is checked).

Criterion for identifying VXLAN packets in hardware:

- UDP destination port is 4789.

What is changed:

- UDP destination port is changed to VXLAN network identifier (VNI) value from VXLAN header.
- Encoding is changed to 2.

Benefits of VXLAN Tunnel Acceleration

- **Generic:** Fewer session resources consumed because we need only the VNI session and not the outer VXLAN UDP session. For VXLAN, we will parse the VXLAN header to extract the VNI and use the VNI to derive a unique flow ID for each VNI within a VXLAN tunnel.
- **With TCI:** VXLAN passthrough traffic will see approximately 30% increase in performance in flow handling with tunnel acceleration compared to the same traffic without tunnel acceleration.
- **Without TCI:** Even without TCI, we will see approximately 10% improvement in performance in flow handling with tunnel acceleration compared to the same traffic without tunnel acceleration. The different flow ID could cause flows to be placed on different dataplanes and thus cause a difference in how the load of a single VXLAN tunnel is distributed for the various VNIs that would be passed in the tunnel. Unless there are VXLAN flows with several VNIs, the performance impact will be mostly negligible.

Disable Tunnel Acceleration

By default, supported firewalls perform [tunnel acceleration](#) to improve performance and throughput for traffic going through GRE tunnels, VXLAN tunnels, and GTP-U tunnels. Tunnel acceleration provides hardware offloading to reduce the time it takes to perform flow lookups and allows the tunnel traffic to be distributed more efficiently based on the inner traffic.

GRE and VXLAN tunnel acceleration is supported on PA-3200 Series firewalls, PA-5450 firewalls, and PA-7000 Series firewalls with PA-7000-100G-NPC-A and PA-7050-SMC-B or PA-7080-SMC-B. You can disable tunnel acceleration to troubleshoot. When you disable tunnel acceleration, you are doing so for GRE, VXLAN, and GTP-U tunnels simultaneously.



There is no performance impact for GRE traffic when disabling tunnel acceleration if no tunnel content inspection (TCI) policies are being used.

STEP 1 | Select **Device > Setup > Management** and edit General Settings.

STEP 2 | Deselect **Tunnel Acceleration** to disable it.

STEP 3 | Click **OK**.

STEP 4 | **Commit**.

STEP 5 | Reboot the firewall.

STEP 6 | (Optional) Verify status of tunnel acceleration.

1. [Access the CLI](#).

2. **> show tunnel-acceleration**

System output is Enabled or Disabled. Additional status and reason for GTP-U only:

- **Disabled**—GTP-U tunnel acceleration is not supported on firewall model or GTP Security is disabled.
- **Error (TCI with GTP-U configured unexpectedly)**—TCI with GTP-U protocol is configured when Tunnel Acceleration is enabled.
- **Enabled**—Tunnel Acceleration is enabled; GTP-U Tunnel Acceleration is not running yet. GTP Security is enabled, but yet to reboot.
- **Installed**—GTP-U Tunnel Acceleration is running.

Network Packet Broker

Network Packet Broker filters and forwards network traffic to an external security chain of one or more third-party security appliances. Network Packet Broker replaces the Decryption Broker feature introduced in PAN-OS 8.1 and expands its capabilities to include forwarding non-decrypted TLS traffic and non-TLS traffic (cleartext) as well as decrypted TLS traffic. The ability to handle all types of traffic is especially valuable in very high security environments such as financial and government institutions.

Network Packet Broker is supported for PA-7000 Series, PA-7000b, PA-5400 Series, PA-5200 Series, PA-3400 Series, and PA-3200 Series devices and VM-300 and VM-700 models. It requires SSL Forward Proxy decryption to be enabled, where the firewall is established as a trusted third party (or man-in-the-middle) to session traffic.



A firewall interface cannot be both a decryption broker and a GRE tunnel endpoint.

- [Network Packet Broker Overview](#)
- [How Network Packet Broker Works](#)
- [Prepare to Deploy Network Packet Broker](#)
- [Configure Transparent Bridge Security Chains](#)
- [Configure Routed Layer 3 Security Chains](#)
- [Network Packet Broker HA Support](#)
- [User Interface Changes for Network Packet Broker](#)
- [Limitations of Network Packet Broker](#)
- [Troubleshoot Network Packet Broker](#)

Network Packet Broker Overview

If you use one or more third-party security appliances (a security chain) as part of your overall security suite, you can use Network Packet Broker to filter and forward network traffic to those security appliances. Network Packet Broker replaces the Decryption Broker feature introduced in PAN-OS 8.1.

Like Decryption Broker, Network Packet Broker provides decryption capabilities and security chain management. This simplifies your network by eliminating complications from supporting dedicated devices for those functions and reduces capital and operating costs. Also like Decryption Broker, Network Packet Broker provides health checks to ensure that the path to the security chain is healthy and options for handling traffic if a chain goes down.

Network Packet Broker expands the firewall's security chain forwarding capabilities so that you can filter and forward not only decrypted TLS traffic, but also non-decrypted TLS and non-TLS (cleartext) traffic to one or more security chains based on applications, users, devices, IP addresses, and zones. These features are especially valuable in very high security environments such as financial and government institutions.

Upgrade and downgrade:

- When you upgrade to PAN-OS 10.2 on firewalls that have a Decryption Broker license:

- The license name changes automatically to Network Packet Broker after you reboot the firewall.



You must reboot the firewall to make the license take effect and update the user interface regardless of whether the firewall is a standalone firewall, part of an HA pair, or if you push Network Packet Broker licenses to firewalls from Panorama.

- PAN-OS translates any existing Decryption Broker Forwarding profiles (**Profiles > Decryption > Forwarding Profile**) into Packet Broker profiles.
- PAN-OS translates any existing Decryption Policy rules for forwarding traffic to security chains into Network Packet Broker policy rules.
- PAN-OS removes the Decryption Broker profile from the user interface and replaces it with the Packet Broker profile (**Profiles > Packet Broker**), and also adds the Network Packet Broker policy (**Policies > Network Packet Broker**).

- When you downgrade to PAN-OS 10.0 from PAN-OS 10.1:

- PAN-OS translates any existing Packet Broker profiles into Decryption Broker Forwarding profiles.

- PAN-OS removes the Network Packet Broker rulebase and prints a warning message. You must reconfigure the Network Packet Broker policy rules as Decryption policy rules for Decryption Forwarding.

- The license name remains Network Packet Broker (the license name changes from Decryption Broker to Network Packet Broker in all PAN-OS versions after a reboot and does not affect the operation of Decryption Broker). However, the functionality is Decryption Broker functionality, not Network Packet Broker functionality.

- PAN-OS removes the Network Packet Broker profile from the user interface and replaces it with the Decryption Forwarding profile, and also removes the Network Packet Broker policy

from the user interface (there is no replacement; you use Decryption Policy rules to forward only decrypted Forward Proxy traffic to security chains).

Requirements for using Network Packet Broker:

- You must install a free Packet Broker license on the firewall. Without the free license, you can't access the Packet Broker policy and profile in the interface.
- The firewall must have at least two available layer 3 Ethernet interfaces to use as a dedicated pair of packet broker forwarding interfaces.
- You can configure multiple pairs of dedicated Network Packet Broker forwarding interfaces to connect to different security chains.
- For each security chain, the pair of dedicated Network Packet Broker interfaces must be in the same security zone.



*Security policy must allow traffic between each paired set of Network Packet Broker interfaces. The **intrazone-default** Security policy rule allows traffic within the same zone by default. However, if you have a "deny all" policy rule earlier in the policy rulebase, then you must create an explicit allow rule to allow the Network Packet Broker traffic.*

- The pair of dedicated interfaces connect to the first and last devices in a security chain.



Network Packet Broker supports routed layer 3 security chains and Transparent Bridge Layer 1 security chains. For routed layer 3 chains, one pair of packet broker forwarding interfaces can connect to multiple layer 3 security chains using a properly configured switch, router, or other device to perform the required layer 3 routing between the firewall and the security chains.

- Dedicated Network Packet Broker forwarding interfaces cannot use dynamic routing protocols.
- None of the devices in the security chain can modify the source or destination IP address, source or destination port, or protocol of the original session because the firewall would not be able to match the modified session to the original session and therefore would drop the traffic.
- You must enable the firewall to **Allow forwarding of decrypted content** (Device > Setup > Content-ID).

Network Packet Broker supports:

- Decrypted TLS, non-decrypted TLS, and non-TLS traffic.
- SSL Forward Proxy, SSL Inbound Inspection, and encrypted SSH traffic.
- Routed layer 3 security chains.
- Transparent Bridge layer 1 security chains.



You can configure both routed layer 3 and layer 1 Transparent Bridge security chains on the same firewall but you must use different pairs of forwarding interfaces for each type.

- Unidirectional traffic flow through the chain: all traffic to the chain egresses the firewall on one dedicated interface and returns to the firewall on another dedicated interface, so all traffic flows in the same direction through the pair of dedicated Network Packet Broker interfaces.
 -  Both firewall forwarding interfaces must be in the same zone.
 - Bidirectional traffic flow through the security chain:
 - Client-to-server (c2s) traffic egresses the firewall on one dedicated firewall broker interface and returns to the firewall on another dedicated firewall broker interface.
 - Server-to-client (s2c) traffic uses the same two dedicated firewall broker interfaces as c2s traffic, but the traffic flows in the opposite direction through the security chain. The firewall broker interface on which the s2c traffic goes to the chain is the same interface on which the c2s traffic returns from the chain to the firewall. The firewall broker interface on which the s2c traffic returns to the firewall is the same interface on which the c2s traffic egresses to the chain.
-  Both firewall forwarding interfaces must be in the same zone.
-  Network Packet Broker does not support multicast, broadcast, or decrypted SSH traffic.

How Network Packet Broker Works

The high-level workflow for connecting the firewall to a chain of third-party security devices is:

1. Identify the non-decrypted TLS, decrypted TLS, and non-TLS (TCP and UDP) traffic to forward.
2. Identify the security chain topology. Determine whether each security chain's devices forward traffic transparently (bridging) or whether the devices route traffic based on Layer 3 information. Using multiple security chains helps load-balance traffic. In addition, decide whether to bypass the security chain (traffic goes through normal processing on the firewall and is forwarded or blocked accordingly) or block the traffic if a security chain fails a health check.
3. Install the free Network Packet Broker license on the firewalls that will forward traffic to the security chain(s).
4. Identify one or more pairs of firewall interfaces to forward traffic to one or more security chains and enable Network Packet Broker on those interfaces.
5. Configure at least one Packet Broker profile.
6. Configure at least one Network Packet Broker policy.

To use a chain of third-party security devices to inspect traffic, you configure three objects on the firewall:

- **Interfaces**—One or more pair of layer 3 Ethernet firewall interfaces for forwarding traffic from the firewall to the security chain and receiving the processed traffic back from the security chain. Configure Network Packet Broker interface pairs before you configure profiles and policy rules because you need to specify the interface pairs in the profiles.
- **Packet Broker profiles**—Profiles control how to forward the traffic that you define in a policy to a security chain. Each Network Packet Broker policy rule has an associated Packet Broker profile. Profiles define whether the security chain is a routed layer 3 chain or a layer 1 Transparent Bridge chain, the direction of traffic through the chain (unidirectional or bidirectional), the dedicated Network Packet Broker firewall interfaces, and how to monitor the health of the connection between the firewall and the security chain. For multiple routed layer 3 security chains, you can specify the first and last device of each chain and a session distribution (load balancing) method for the associated traffic.
- **Network Packet Broker policy rules**—Policy rules define the application traffic to forward to each security chain or to load balance for multiple routed (layer 3) chains. Policy rules define the source and destination, users, applications, and services of traffic to forward to a security chain. Policy rules also define the type of traffic to forward to a security chain: you can select decrypted TLS traffic, non-decrypted TLS traffic, non-TLS traffic, or any combination of traffic types. You also add a Packet Broker profile in each policy rule to specify the security chain to which to forward traffic (and all of the other profile characteristics).

Use [Policy Optimizer](#) to review and tighten Network Packet Broker policy rules.

To match application traffic to Network Packet Broker policy rules, Network Packet Broker looks up applications in the firewall App-ID cache. If the application is not in the App-ID cache, then the firewall bypasses the security chain and applies any threat inspection that is configured in the Security policy allow rule to the traffic. If the application is in the App-ID cache, then the firewall forwards the traffic to the security chain in the manner specified by the Network Packet Broker policy rule and its associated Packet Broker profile.

For non-decrypted TLS and non-TLS traffic, the firewall installs the application in the App-ID cache on the first session, so the firewall treats the traffic as specified in the Network Packet Broker policy and profile.

For decrypted TLS traffic, on the *first session* for an application, Network Packet Broker doesn't know that the session is being decrypted and sees "ssl" as the application. The underlying specific application is not yet known or installed in the App-ID cache, so the broker lookup fails and the traffic bypasses the security chain. The traffic is still subject to any threat inspection configured on the Security policy allow rule. When the firewall decrypts the traffic, the firewall learns the specific application and installs it in the App-ID cache. For the second and subsequent decrypted sessions for the same application, Network Packet Broker lookups succeed because the specific application is now in the App-ID cache, and the firewall forwards traffic to the security chain as expected.

Prepare to Deploy Network Packet Broker

Take the following actions to prepare to deploy Network Packet Broker:

1. Obtain and activate the free Network Packet Broker license.
 1. Log in to the [Customer Support Portal](#).
 2. Select **Assets > Devices** on the left-hand navigation pane.
 3. Find the device on which you want to enable decryption broker or decryption port mirroring and select **Actions** (the pencil icon).
 4. Under Activate Licenses, select **Activate Feature License**
 5. Select the **Network Packet Broker** free license.
 6. Click **Agree and Submit**.
2. Install the license on the firewall.
 1. Select **Device > Licenses**.
 2. Click **Retrieve license keys from the license server**.
 3. Verify that the **Device > Licenses** page shows that the **Network Packet Broker** license is now active on the firewall.
 4. Restart the firewall (**Device > Setup > Operations**). Network Packet Broker is not available for configuration until the firewall restarts.



You can push the Network Packet Broker license from Panorama to managed firewalls. You must reboot the firewalls to make the license take effect and update the user interface.

3. Enable the App-ID cache for Network Packet Broker.

1. The App-ID cache is disabled by default. Enable it using the configuration mode CLI command:

```
admin@PA-3260# set deviceconfig setting application cache yes
```

2. Enable the firewall to use the App-ID cache to identify applications:

```
admin@PA-3260# set deviceconfig setting application use-cache-for-identification yes
```

Verify the settings show that Application cache is set to yes and Use cache for appid is set to yes:

```
admin@PA-3260> show running application setting
Application setting:
Application cache          : yes
Supernode                  : yes
Heuristics                 : yes
Cache Threshold            : 1
Bypass when exceeds queue limit: no
Traceroute appid           : yes
```

Traceroute TTL threshold	:	30
Use cache for appid	:	yes
Use simple appsigs for ident	:	yes
Use AppID cache on SSL/SNI	:	no
Unknown capture	:	on
Max. unknown sessions	:	5000
Current unknown sessions	:	33
Application capture	:	off

Current APPID Signature	:	
Memory Usage	:	16768 KB (Actual 16461 KB)
TCP 1 C2S	:	regex 11898 states
TCP 1 S2C	:	regex 4549 states
UDP 1 C2S	:	regex 4263 states
UDP 1 S2C	:	regex 1605 states

4. Enable the firewall to **Allow forwarding of decrypted content** (Device > Setup > Content-ID).
5. Identify the traffic that you want to forward to one or multiple security chains.
6. Identify the topology for each security chain and determine whether to use layer 1 Transparent Bridge forwarding or routed layer 3 forwarding, which determines what type of security chain you configure on the firewall. Considerations include:
 - Whether you want to load-balance traffic across multiple chains (use a routed layer 3 security chain to distribute sessions across multiple chains through a router, switch, or other routing device), use a single chain, or use different security chains for different types of traffic. For multiple layer 1 Transparent Bridge chains, you need a pair of dedicated firewall interfaces for each security chain because the layer 1 connection is not routed.
 - Whether to use unidirectional or bidirectional traffic flow through the security chain.
7. Decide which pairs of firewall interfaces to use as dedicated Network Packet Broker forwarding interfaces.
 - For layer 1 Transparent Bridge chains, you need a pair of dedicated firewall interfaces for each layer 1 security chain. You can configure policy rules to send specific traffic to different security chains.
 - For routed layer 3 chains, one dedicated pair of firewall interfaces can load balance traffic among multiple layer 3 security chains through a switch, router, or other routing-capable device.
 - For routed layer 3 chains, you can use multiple pairs of dedicated firewall interfaces to send specific traffic to different security chains using different policy rules.



*Security policy must allow traffic between each paired set of Network Packet Broker interfaces. The **intrazone-default** Security policy rule allows traffic within the same zone by default. However, if you have a “deny all” policy rule earlier in the policy rulebase, then you must create an explicit allow rule to allow the Network Packet Broker traffic.*

Configure Transparent Bridge Security Chains

A layer 1 Transparent Bridge security chain forwards traffic from one firewall interface through a directly connected series of data inspection and processing security devices and then back through a different firewall interface without the need to route the traffic.

Before you configure a layer 1 Transparent Bridge security chain, take the steps to [Prepare to Deploy Network Packet Broker](#), including ensuring that the physical connections between the firewall and the security chain devices are correct and that you allow the firewall to forward decrypted content.

To distribute sessions across multiple Transparent Bridge security chains, create one layer 1 Transparent Bridge security chain on the firewall for each of the security chains you want to use to load balance traffic. Each Transparent Bridge security chain on the firewall requires two dedicated layer 3 Ethernet interfaces. Check to ensure that you have enough free Ethernet interfaces for the topology you want to configure.



Layer 1 Transparent Bridge security chains cannot failover to another security chain because they are not routed.

STEP 1 | Enable two Layer 3 Ethernet interfaces as Network Packet Broker forwarding interfaces.

1. Select **Network > Interfaces > Ethernet**.
2. Select an unused Ethernet interface to use as one of the two Network Packet Broker forwarding interfaces.
3. Set the **Interface Type** to **Layer3**.
4. (Optional) To use Health Monitoring on Transparent Bridge paths, you must configure a static IP address on each interface. Select **IPv4** and **Add** a static Layer 4 IP address. You can

also configure IPv6 addresses for Transparent Bridge interfaces (but not for Routed Layer 3 security chains).

5. On the **Config** tab, select a zone to assign the interface to.



You must configure both of the security chain interfaces in the same zone.

*Security policy must allow traffic between each paired set of Network Packet Broker interfaces. The **intrazone-default** Security policy rule allows traffic within the same zone by default. However, if you have a “deny all” policy rule earlier in the policy rulebase, then you must create an explicit allow rule to allow the Network Packet Broker traffic.*

6. On the **Config** tab, as a best practice, use or create a dedicated Virtual Router to assign the interface to. Using a dedicated Virtual Router ensures that the Network Packet Broker interface traffic remains separate from other traffic.
7. Select **Advanced** and then select **Network Packet Broker** to enable the interface.



*If you inadvertently configured a DHCP or PPPoE address, the **Network Packet Broker** option is grayed out. Return to the **IPv4** tab and set the **Type** to **Static**.*

8. Click **OK** to save the interface configuration.
9. Repeat this procedure on another unused Ethernet interface to configure the other Network Packet Broker forwarding interface.

STEP 2 | Configure a Packet Broker profile to control how to forward the traffic to the layer 1 Transparent Bridge security chain.

1. Select **Objects > Packet Broker Profile** and **Add** a new profile or modify an existing profile.
2. Give the profile a **Name** and **Description** so that you easily identify its purpose.
3. On the **General** tab:
 - Select **Transparent Bridge (Layer 1)** as the **Security Chain Type**.
 - **Enable IPv6** if the traffic is IPv6 traffic.
 - Select the **Flow Direction**.



Your network topology determines whether to use unidirectional or bidirectional flows. The performance is approximately the same using either method.

To use one firewall interface to forward both the c2s and s2c session flows to the security chain and use the other firewall interface to receive both session flows back from the security chain, select **Unidirectional**.

To use **Interface #1** to forward the c2s flow to the security chain and receive the s2c flow from the security chain, and use **Interface #2** to forward the s2c flow to the security chain and receive the c2s flow from the security chain, select **Bidirectional**.

- Specify the Network Packet Broker forwarding interface pair in **Interface #1** and **Interface #2**. Both interfaces must already be enabled for Network Packet Broker (see [Prepare to Deploy Network Packet Broker](#)) to be available for use. Be careful to pay

attention to the directionality of flow when you configure which interface is **Interface #1** and which interface is **Interface #2**.

The screenshot shows the 'Packet Broker Profile' configuration window with the 'General' tab selected. Key settings include:

- Name:** User Traffic Security Chain
- Description:** Traffic chain to inspect common user traffic
- Security Chain Type:** Transparent Bridge (Layer 2)
- Enable IPv6:** Unchecked
- Flow Direction:** Bidirectional (selected)
- Client-to-Server flow via Interface #1:** (disabled)
- Server-to-Client flow via Interface #2:** (disabled)
- Interface #1:** ethernet1/10
- Interface #2:** ethernet1/11

Buttons at the bottom right include 'OK' and 'Cancel'.

4. The **Security Chains** tab is not used for Transparent Bridges.

5. On the **Health Monitor** tab:

- Select the type or types of health monitoring you want to perform so that you can control what happens if the security chain experiences a failure. You can select one, two, or all from **Path Monitoring**, **HTTP Monitoring**, and **HTTP Monitoring Latency**.

Path Monitoring—Checks device connectivity using pings.

HTTP Monitoring—Checks device availability and response time.

HTTP Monitoring Latency—Checks device processing speed and efficiency. When you select this option, **HTTP Monitoring** is automatically enabled as well.

- Enabling one or more types of health monitoring activates the **On Health Check Failure** options, which determine how the firewall handles security chain traffic if there is a security chain health failure. The options are **Bypass Security Chain** and **Block Session**.

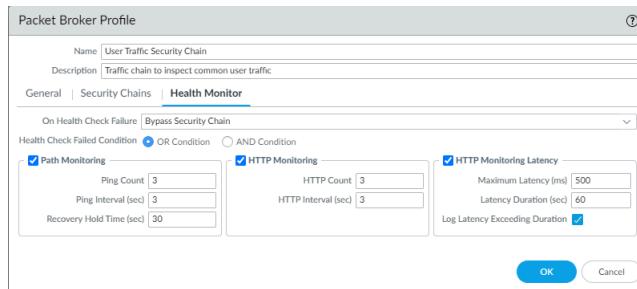
Bypass Security Chain—The firewall forwards the traffic to its destination instead of to the security chain and applies any configured Security profiles and protections to the traffic.

Block Session—The firewall blocks the session.

The method you select depends on how you want to treat the traffic if you can't run the traffic through the security chain.

- If you select more than one health check option, select whether you want the firewall to consider the health check as failed (**Health Check Failed Condition**) if any one of the monitoring options records a failed condition (**OR Condition**) or only if all of the selected monitoring options record a failed condition (**AND Condition**). For example, if you enable all three health check options and one of the options records a failed condition, if you selected **OR Condition**, the firewall considers the security chain connection to be failed and executes the action you specified in **On Health Check Failure**. If you selected **AND**

Condition, the firewall would still consider the connection to be healthy because two of the health metrics are still OK.

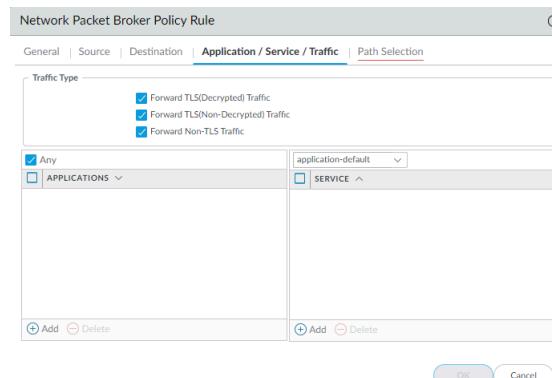


- Click **OK** to save the profile.

STEP 3 | Configure a Packet Broker policy to define the traffic to forward to the layer 1 Transparent Bridge security chain.

- Select **Policies > Network Packet Broker** and **Add** a new policy rule or modify an existing policy rule.
- On the **General** tab, give the policy rule a **Name** and **Description** so that you easily identify its purpose, add an **Audit Comment**, and apply tags if you use them.
- On the **Source** tab, identify the source zones, IP addresses, users, and devices of the traffic that you want the rule to forward to the security chain.
- On the **Destination** tab, identify the destination zones, IP addresses, and devices of the traffic that you want the rule to forward to the security chain.
- On the **Application/Service/Traffic** tab, identify the applications and services you want the rule to forward to the security chain. Unless the rule controls applications that you expect to use non-standard ports such as internal custom applications, the best practice is to set the **Service to Application Default** so that applications that exhibit evasive behavior by using non-standard ports are blocked.

For the **Traffic Type**, select all of the types of traffic that you want the rule to forward to the security chain. **Forward TLS(Decrypted) Traffic** is the default selection. You can select any combination of **Forward TLS(Decrypted) Traffic**, **Forward TLS(Non-Decrypted)**, and **Forward Non-TLS Traffic** to forward to the security chain.



- On the **Path Selection** tab, select the Packet Broker profile you created in [Step 2](#) or create a new profile to control how to send the traffic that the policy rule controls to the security chain.

STEP 4 | Repeat [Step 1](#) through [Step 3](#) to create more layer 1 Transparent Bridge security chains.

For each layer 1 Transparent Bridge security chain:

- The two Ethernet interfaces used as Network Packet Broker forwarding interfaces must be dedicated to each security chain. Ethernet interfaces used for a Transparent Bridge security chain cannot be used for any other purpose or carry any other traffic.
- Each pair of Network Packet Broker forwarding interfaces connects to one layer 1 Transparent Bridge security chain.

You can load balance traffic by creating Network Packet Broker policy rules that divide traffic relatively equally among the Transparent Bridge security chains. You can also use policy rules to direct specific traffic and types of traffic through specific security chains.



*Layer 1 Transparent Bridge security chains cannot failover to another security chain because they are not routed. Use the **Health Monitor** tab in the Packet Broker profile to configure how to handle traffic if a Transparent Bridge security chain fails.*

Configure Routed Layer 3 Security Chains

A routed layer 3 security chain forwards traffic to a series of data inspection and processing security devices and then back to the firewall using two dedicated forwarding interfaces on the firewall.

Before you configure a routed layer 3 security chain, take the steps to [Prepare to Deploy Network Packet Broker](#), including ensuring that the physical connections between the firewall and the security chain devices are correct and that you allow the firewall to forward decrypted content. Check to ensure that you have enough free Ethernet interfaces on the firewall for the topology you want to configure.

Each routed layer 3 security chain that you configure on the firewall requires two dedicated layer 3 Ethernet interfaces, which can connect to one layer 3 security chain or distribute sessions (load balance) to up to 64 layer 3 security chains with a properly configured router, switch, or similar device between the firewall and the security chains.



Network Packet Broker cannot forward IPv6 traffic on a routed layer 3 security chain. To forward IPv6 traffic, use a Transparent Bridge (layer 1) security chain.

STEP 1 | Enable two Layer 3 Ethernet interfaces as Network Packet Broker forwarding interfaces.

1. Select **Network > Interfaces > Ethernet**.
2. Select an unused Ethernet interface to use as one of the two Network Packet Broker forwarding interfaces.
3. Set the **Interface Type** to **Layer3**.
4. On the **Config** tab, select a zone to assign the interface to.



You must configure both of the security chain interfaces in the same zone.

Security policy must allow traffic between each paired set of Network Packet Broker interfaces. The **intrazone-default** Security policy rule allows traffic within the same zone by default. However, if you have a "deny all" policy rule earlier in the policy rulebase, then you must create an explicit allow rule to allow the Network Packet Broker traffic.

5. On the **Config** tab, as a best practice, use or create a dedicated Virtual Router to assign the interface to. Using a dedicated Virtual Router ensures that the Network Packet Broker interface traffic remains separate from other traffic.
6. Select **Advanced** and then select **Network Packet Broker** to enable the interface.

The screenshot shows the 'Ethernet Interface' configuration dialog box. The 'Advanced' tab is selected. In the 'Link Settings' section, there is a checkbox labeled 'Network Packet Broker' which is checked and highlighted with a yellow box. Other settings like Link Speed, MTU, and Adjust TCP MSS are also visible.

7. Click **OK** to save the interface configuration.
8. Repeat this procedure on another unused Ethernet interface to configure the other Network Packet Broker forwarding interface.

STEP 2 | Configure a Packet Broker profile to control how to forward the traffic to the routed layer 3 security chain.

1. Select **Objects > Packet Broker Profile** and **Add** a new profile or modify an existing profile.
2. Give the profile a **Name** and **Description** so that you easily identify its purpose.
3. On the **General** tab:
 - Select **Routed (Layer 3)** as the **Security Chain Type**.
 - Select the **Flow Direction**.



Your network topology determines whether to use unidirectional or bidirectional flows. The performance is approximately the same using either method.

To use one firewall interface to forward both the c2s and s2c session flows to the security chain and use the other firewall interface to receive both session flows back from the security chain, select **Unidirectional**.

To use **Interface #1** to forward the c2s flow to the security chain and receive the s2c flow from the security chain, and use **Interface #2** to forward the s2c flow to the security chain and receive the c2s flow from the security chain, select **Bidirectional**.

- Specify the Network Packet Broker forwarding interface pair in **Interface #1** and **Interface #2**. Both interfaces must already be enabled for Network Packet Broker (see [Step 1](#)) to be available for use. Be careful to pay attention to the directionality of flow when you configure which interface is **Interface #1** and which interface is **Interface #2**.

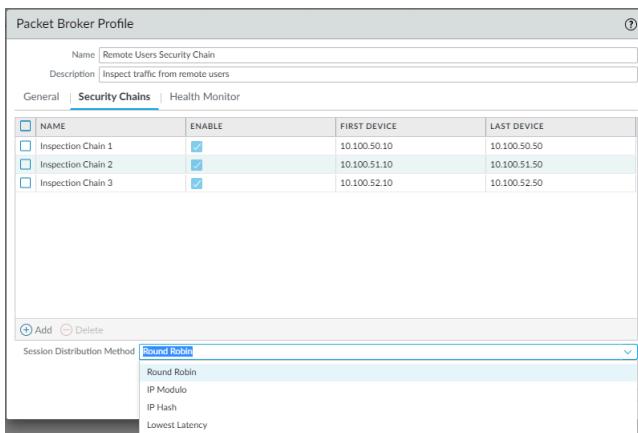


Session distribution (load balancing) only applies to new sessions. The firewall does not rebalance traffic in the middle of a session. The firewall only distributes sessions to security chains whose status is "up" (active, healthy).

4. On the **Security Chains** tab, **Add** the IP addresses of the first and last device in each routed layer 3 security chain to which you want to connect. You must specify at least one security chain or the firewall can't route traffic to a chain and you can't save the profile.

If you specify multiple routed layer 3 security chains, then you also need to place a correctly configured router, switch, or similar device between the firewall and the security chains to

perform the proper routing. In addition, specify the **Session Distribution Method** to load balance the traffic among the security chains.



5. On the **Health Monitor** tab:

- Select the type or types of health monitoring you want to perform so that you can control what happens if the security chain experiences a failure.

You can select one, two, or all from **Path Monitoring**, **HTTP Monitoring**, and **HTTP Monitoring Latency**.

Path Monitoring—Checks device connectivity using pings.

HTTP Monitoring—Checks device availability and response time.

HTTP Monitoring Latency—Checks device processing speed and efficiency. When you select this option, **HTTP Monitoring** is automatically enabled as well.

- Enabling one or more types of health monitoring activates the **On Health Check Failure** options, which determine how the firewall handles security chain traffic if there is a security chain health failure.

If you configure multiple security chains on one set of routed layer 3 Network Packet Broker interfaces, then on a security chain failure, traffic fails over to the remaining healthy security chains. If there is no security chain available to handle failover traffic, the firewall takes the action configured **On Health Check Failure**. The options are **Bypass Security Chain** and **Block Session**.

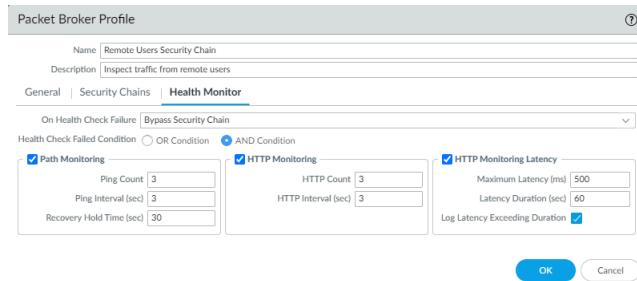
Bypass Security Chain—The firewall forwards the traffic to its destination instead of to the security chain and applies any configured Security profiles and protections to the traffic.

Block Session—The firewall blocks the session.

The method you select depends on how you want to treat the traffic if you can't run the traffic through the security chain.

- If you select more than one health check option, select whether you want the firewall to consider the health check as failed (**Health Check Failed Condition**) if any one of the monitoring options records a failed condition (**OR Condition**) or only if all of the selected monitoring options record a failed condition (**AND Condition**). For example, if you enable all three health check options and one of the options records a failed condition, if you selected **OR Condition**, the firewall considers the security chain connection to be failed

and executes the action you specified in **On Health Check Failure**. If you selected **AND Condition**, the firewall would still consider the connection to be healthy because two of the health metrics are still OK.



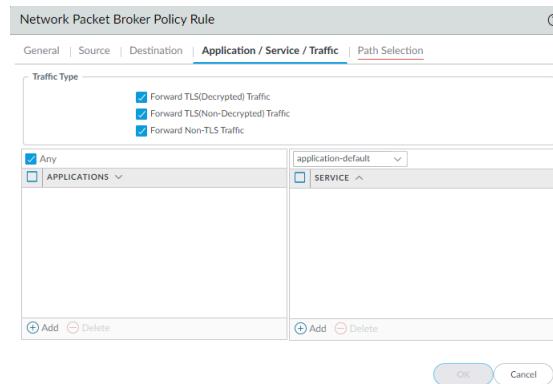
6. Click **OK** to save the profile.

STEP 3 | Configure a Packet Broker policy to define the traffic to forward to the routed layer 3 security chain.

1. Select **Policies > Network Packet Broker** and **Add** a new policy rule or modify an existing policy rule.
2. On the **General** tab, give the policy rule a **Name** and **Description** so that you easily identify its purpose, add an **Audit Comment**, and apply tags if you use them.
3. On the **Source** tab, identify the source zones, IP addresses, users, and devices of the traffic that you want the rule to forward to the security chain.
4. On the **Destination** tab, identify the destination zones, IP addresses, and devices of the traffic that you want the rule to forward to the security chain.
5. On the **Application/Service/Traffic** tab, identify the applications and services you want the rule to forward to the security chain. Unless the rule controls applications that you expect to use non-standard ports such as internal custom applications, the best practice is to set the **Service to Application Default** so that applications that exhibit evasive behavior by using non-standard ports are blocked.

For the **Traffic Type**, select all of the types of traffic that you want the rule to forward to the security chain. **Forward TLS(Decrypted) Traffic** is the default selection. You can select

any combination of **Forward TLS(Decrypted) Traffic**, **Forward TLS(Non-Decrypted)**, and **Forward Non-TLS Traffic** to forward to the security chain.



6. On the **Path Selection** tab, select the Packet Broker profile you created in [Step 2](#) or create a new profile to control how to send the traffic that the policy rule controls to the security chain.

STEP 4 | If you want to create separate routed layer 3 security chains that use different dedicated pairs of firewall interfaces, then repeat [Step 1](#) through [Step 3](#) to create more Network Packet Broker security chains. The two layer 3 Ethernet interfaces used as Network Packet Broker forwarding interfaces must be dedicated to the security chain and cannot be used for any other purpose or carry any other traffic.

Network Packet Broker HA Support

In addition to the path and latency health monitoring available in the Packet Broker profile to protect against security chain failures, you can also configure [High Availability \(HA\)](#) on firewalls that have Network Packet Broker forwarding interfaces to protect against firewall failures. Configuring both path monitoring and HA protects not only against security chain failures but also against firewall failures.

Network Packet Broker supports Active/Passive HA pairs. Active/Active HA pairs are not supported because the dedicated broker forwarding interfaces must be specified in the Packet Broker profile.

After a failover, decrypted SSL traffic is reset because SSL state isn't synchronized between HA nodes. Cleartext traffic resumes if the session is correctly synchronized and the TCP sequence is correctly relearned.

User Interface Changes for Network Packet Broker

Network Packet Broker replaces the Decryption Broker feature introduced in PAN-OS 8.1 and expands its capabilities to include forwarding non-decrypted TLS and non-TLS traffic as well as decrypted TLS traffic to a security chain. To support Network Packet Broker, the PAN-OS 10.2 user interface has the following changes:

- A new policy (**Policies > Network Packet Broker**) enables you to configure the specific traffic to forward to the security chain and attach a Packet Broker profile to control how to forward the specified traffic to the security chain.
-  *Decryption Broker used Decryption policy rules to forward only decrypted TLS traffic to the security chain. The new Network Packet Broker policy rules enable you to select not only decrypted TLS traffic, but also encrypted TLS traffic and non-TLS traffic.*
- A new profile (**Objects > Packet Broker Profile**) replaces the old **Objects > Decryption > Decryption Broker Profile** and enables you to configure exactly how to forward traffic to the security chain and monitor path and latency health. On the **General** tab, the names of the fields where you enter the dedicated firewall Network Packet Broker forwarding interface pair changed from “Primary Interface” and “Secondary Interface” to **Interface #1** and **Interface #2**, respectively.
 - When you select **Policies > Network Packet Broker**, you can then select any of the **Rule Usage** options in **Policy Optimizer** to view Network Packet Broker policy usage information. **Rule Usage** statistics help you evaluate whether you need to keep unused Network Packet Broker rules or if you can delete them and tighten up the rulebase to reduce the attack surface.
 - Because Network Packet Broker replaced Decryption Broker, Decryption policy no longer handles brokering traffic to a security chain. For that reason, on the **Options** tab, the **Decrypt and Forward** option is no longer an **Action** that the policy can take, and the **Forwarding Profile** field was also removed because now only Decryption profiles are valid on Decryption policies.
 - In **Network > Interfaces > Ethernet**, when you set the **Interface Type** to Layer 3 and then select the **Advanced** tab, the name of the checkbox to enable the interface as forwarding interface for Network Packet Broker changed from “Decrypt Forward” to **Network Packet Broker**.
 - For **Device > Admin Roles**, on the **Web UI** tab, there are two changes:
 - Under **Policies**, you can now configure **Network Packet Broker** admin role permissions.
 - Under **Objects**, the **Decryption > Forwarding Profile** option is removed and replaced by the **Packet Broker Profile** option for admin role permissions.
 - On firewalls, for **Monitor > Manage Custom Reports**, when you select **Traffic Log** from the **Detailed Logs** as the **Database**, in the **Available Columns** list, you can now select **Forwarded to Security Chain**.
- On Panorama, for **Monitor > Manage Custom Reports**, when you select **Panorama Traffic Log** from the **Detailed Logs** as the **Database**, in the **Available Columns** list, you can now select **Forwarded to Security Chain**.
- In the Traffic log, the “Decrypt Forward” column is renamed **Forwarded to Security Chain**. In the detailed view of the Traffic log, in the **Flags** section, the checkbox “Decrypt Forwarded” is renamed to **Forwarded to Security Chain**.

- The free license for the feature is renamed from “Decryption Broker” to **Packet Broker**. If you have the free Decryption Broker license on your firewall, the name changes automatically when you upgrade to PAN-OS 10.1. The change is only in the name and has no effect on the feature.

Limitations of Network Packet Broker

Most Palo Alto Networks platforms support Network Packet Broker, but a few do not and a few have some limits:

- Support is not available in Prisma Access or in NSX.
- AWS, Azure, and GCP only support routed layer 3 security chains.

Network Packet Broker has a few limitations on Panorama for managed firewalls and a few usage limitations. On Panorama:

- If you push Network Packet Broker licenses to managed firewalls, you must reboot the firewalls for the licenses and the associated user interface elements to be installed.
- You cannot create a Packet Broker profile in a **Shared** context because you configure specific interfaces in the Packet Broker profile.
- Different Device Groups cannot share the same Packet Broker profiles.
- Panorama cannot push a Network Packet Broker configuration (Network Packet Broker policy rules and profiles) to a Device Group that contains firewalls which run a PAN-OS version older than 10.1.

If you want to use Network Packet Broker in a Device Group that contains firewalls on multiple PAN-OS versions and some of those firewalls run a PAN-OS version older than 10.1, then you must either upgrade the pre-10.2 firewalls to PAN-OS 10.2 or remove the pre-10.2 firewalls from the Device Group before you push the Network Packet Broker configuration.



You can use Panorama to push a Packet Broker profile that is attached to a Decryption policy rule to pre-10.1 firewalls that have Decryption Broker licenses installed. The Action for the rule (Options tab) must be Decrypt and Forward and you must attach the Packet Broker profile to the rule (Decryption Profile setting on the Options tab). Pre-10.2 firewalls use the Packet Broker profile as the Decryption Forwarding profile for Decryption Broker. The Decryption policy rule determines the traffic to which the firewall applies the profile.

The traffic that the Decryption policy rule controls must be decrypted SSL traffic (Decryption Broker doesn't support encrypted SSL traffic or cleartext traffic).

- When you upgrade from PAN-OS 10.0 to PAN-OS 10.1, only local Decryption policy rules that are used for Decryption Broker are migrated to Network Packet Broker rules. Decryption Broker policy rules that were pushed from Panorama to firewalls are migrated automatically on Panorama but are not migrated automatically on the firewall. Decryption Broker policy rules configured locally on a firewall are migrated to Network Packet Broker rules on that firewall only. For rules configured on Panorama, Panorama must do another commit push to the firewall to synchronize the Decryption Broker rules that were migrated to Network Packet Broker rules on Panorama.
- When you downgrade from PAN-OS 10.2 to PAN-OS 10.0, Network Packet Broker rules are removed automatically.

Network Packet Broker also has a few usage limitations:

- If the Network Packet Broker firewall also performs source network address translation (SNAT) and the traffic is cleartext traffic, then the firewall performs NAT on the traffic and forwards

the traffic to the security chain. The security chain appliances only see NAT addresses, not the original source addresses:

1. The firewall performs NAT on the client's traffic.
2. The firewall forwards the traffic to the security chain and any routing must be based on the NAT address.
3. Because the source address in the packet is now the NAT address, the security chain appliances only see the NAT address. They do not see the actual client source address.
4. When the security chain returns the traffic to the firewall, the result is that the firewall doesn't know who the user is.

You can find out who the source user was for a session by checking the Traffic logs for that session and correlating the packet with those logs. Traffic logs include both the original source address, from which you can determine the source user, and the SNAT address.



You can avoid this scenario by performing NAT on a device other than the firewall.

- Decrypted SSH, multicast, and broadcast traffic are not supported.
- Client authentication is not supported for SSL Inbound Inspection when RSA certs are used.
- In layer 1 Transparent Bridge mode, if a security chain fails, there's no failover because when you use Transparent Bridge connections, each pair of dedicated Network Packet Broker firewall interfaces connect to one security chain only. (You can't route traffic on layer 1, you can only forward it to the next connected device.)
- You can forward IPv6 traffic only in layer 1 Transparent Bridge mode. You cannot forward IPv6 traffic in Routed (layer 3) mode.
- You cannot use tunnel or loopback interfaces as Network Packet Broker interfaces.
- Network Packet Broker interfaces cannot use dynamic routing protocols.
- Both interfaces must be in the same zone.
- Devices in a security chain cannot modify the source IP address, destination IP address, source port, destination port, or protocol of the original session because the firewall would be unable to match the modified session to the original session and therefore would drop the traffic
- High Availability for Network Packet Broker is supported only for Active/Passive HA firewall pairs. High Availability for Network Packet Broker is not supported for Active/Active firewall pairs.
- High Availability is not supported for SSL traffic. SSL Sessions reset on failovers.
- When you upgrade from PAN-OS 10.0 to PAN-OS 10.1, local Decryption policy rules that are used for Decryption Broker are migrated to Network Packet Broker rules.
- When you downgrade from PAN-OS 10.2 to PAN-OS 10.0, Network Packet Broker rules are removed automatically.

Troubleshoot Network Packet Broker

If you encounter issues configuring Network Packet Broker, check the following items:

- Firewall configuration:
 - Check the next-hop route on the forwarding interface pairs to ensure that it specifies the correct device interface.
 - IP addresses of the chain devices and the firewall interfaces and ensure that they are properly entered in the Packet Broker profile.
 - If HA is enabled, check that the correct interfaces are specified in the profile.
 - Check the flow direction of traffic through the chain.
 - Ensure that the profile indicates the appropriate security chain type.
- Security chain configuration; check:
 - IP addresses, next-hop addresses, and default gateways for each appliance in the security chain.
 - The configuration of any devices between the firewall and the security chain (routers, switches, etc.) for IP addressing, next-hop, and default gateway misconfiguration.
 - The path between the firewall and the chain.
- Check firewall Traffic logs to validate that you see the “Forwarded” flag set as expected for brokered traffic.
- Useful CLI commands include:
 - `show rulebase network-packet-broker`
 - `show running network-packet-broker status`
 - `show running network-packet-broker statistics`
 - `show running application-cache all`
 - `show running application setting`—Confirm that the App-ID cache is enabled and that the cache is used for App-ID, check the cache threshold setting, etc.

Advanced Routing

PAN-OS® 10.2 provides an Advanced Routing Engine that allows the firewall to scale and provide stable, high-performing, and highly available routing functions to large data centers, ISPs, enterprises, and cloud users. The Advanced Routing Engine simplifies operations with a standards-based configuration, which reduces your learning curve since it is similar to that of other router vendors. Protocol configuration profiles and a granular filtering profile work across multiple logical routers and virtual systems. Route redistribution is simplified with a redistribution profile. BGP peer groups and peers can inherit configuration to make BGP more agile.

The Advanced Routing Engine supports static routes, BGP, MP-BGP, OSPFv2, OSPFv3, RIPv2, IPv4 multicast routing, BFD, redistribution, route filtering into the RIB, access lists, prefix lists, and route maps.

Use the [Advanced Routing Engine Migration Reference](#) to plan your migration from the legacy routing engine and to see the differences between the legacy and advanced routing engines and the exceptions.

The following models support the Advanced Routing Engine:

- PA-7000 Series
- PA-5400 Series
- PA-5200 Series
- PA-3400 Series
- PA-3200 Series
- PA-400 Series
- VM-Series
- M-700 appliance
- M-600 appliance
- M-500 appliance
- M-300 appliance
- M-200 appliance

Learn about advanced routing profiles and perform the following tasks to configure advanced routing:

- [Enable Advanced Routing](#)
- [Logical Router Overview](#)
- [Configure a Logical Router](#)
- [Create a Static Route](#)
- [Configure BGP on an Advanced Routing Engine](#)
- [Create BGP Routing Profiles](#)
- [Create Filters for the Advanced Routing Engine](#)

- [Configure OSPFv2 on an Advanced Routing Engine](#)
- [Create OSPF Routing Profiles](#)
- [Configure OSPFv3 on an Advanced Routing Engine](#)
- [Create OSPFv3 Routing Profiles](#)
- [Configure RIPv2 on an Advanced Routing Engine](#)
- [Create RIPv2 Routing Profiles](#)
- [Create BFD Profiles](#)
- [\(PAN-OS 10.2.2 and later 10.2 releases\) Configure IPv4 Multicast](#)
- [\(PAN-OS 10.2.2 and later 10.2 releases\) Create Multicast Routing Profiles](#)
- [\(PAN-OS 10.2.2 and later 10.2 releases\) Create an IPv4 MRoute](#)

Enable Advanced Routing

Although a supported firewall can have a configuration that uses the legacy routing engine and a configuration that uses the Advanced Routing Engine, only one routing engine is in effect at a time. Each time you change the engine that the firewall will use (you enable or disable Advanced Routing to access the advanced engine or legacy engine, respectively), you must commit the configuration and reboot the firewall for the change to take effect.



Before you switch to the Advanced Routing Engine, make a backup of your current configuration.

Similarly, if you configure Panorama with a template that enables or disables Advanced Routing, after you commit and push the template to devices, you must reboot the devices in the template for the change to take effect.



When configuring Panorama, create device groups and Templates for devices that all use the same Advanced Routing setting (all enabled or all disabled). Panorama won't push configurations with Advanced Routing enabled to lower-end firewalls that don't support Advanced Routing. For those firewalls, Panorama will push a legacy configuration if one is present.

The Advanced Routing Engine supports multiple logical routers (known as virtual routers on the legacy routing engine). The number of logical routers supported depends on the firewall model and is the same as the number of virtual routers supported on the legacy routing engine. The Advanced Routing Engine has more convenient menu options and there are many settings that you can easily configure in a profile (authentication, timers, address family, or redistribution profile) that you apply to a BGP peer group or peer, for example. There are also many static route, OSPF, OSPFv3, RIPv2, multicast, and BFD settings on the Advanced Routing Engine.

The Advanced Routing Engine supports RIB filtering, which means you can create a route map to match static routes or routes received from other routing protocols and thus filter which routes are installed in the RIB for the logical router. This function is useful on firewalls with a smaller RIB or FIB capacity; you can still propagate the necessary routing updates without using memory needed elsewhere.

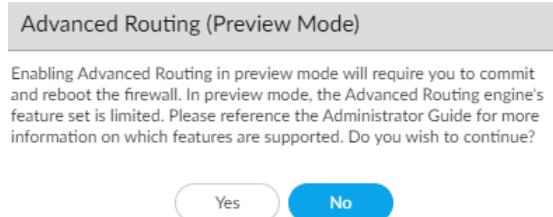
STEP 1 | Make a backup of your current configuration before you enable Advanced Routing.

STEP 2 | Enable the Advanced Routing Engine.

1. Select **Device > Setup > Management** and edit the General Settings.
2. Enable Advanced Routing.

The screenshot shows the 'General Settings' configuration page. It includes fields for Hostname (VM-17-233), Domain, and Login Banner (VM-17-233). Under SSL/TLS Service Profile, 'None' is selected. Time Zone is set to US/Pacific, Locale to en, Date to 2021/06/09, and Time to 17:33:34. Latitude and Longitude fields are present. A section of checkboxes at the bottom includes: 'Automatically Acquire Commit Lock' (unchecked), 'Certificate Expiration Check' (unchecked), 'Use Hypervisor Assigned MAC Addresses' (checked), 'GTP Security' (unchecked), 'SCTP Security' (unchecked), 'Advanced Routing' (checked and highlighted with a yellow box), and 'Tunnel Acceleration' (checked).

3. Before you click OK, make sure you have made a backup of your configuration for the legacy routing engine.
4. Click **OK**.
5. (**PAN-OS 10.2.0 to 10.2.3**) A message about preview mode appears; click **Yes** to proceed to Commit step.



6. (**PAN-OS 10.2.4 and later 10.2 releases**) A warning appears:

Warning



Enabling Advanced Routing will require you to migrate your configuration, **commit** your configuration and, **reboot** the firewall.

If you select **Yes**, a script will assist you in migrating your existing configuration to the Advanced Routing Engine. The migration tool will convert each Virtual Router to a Logical Router.
If you select **Skip**, the system changes to Advance Routing mode without any Logical Router configuration.

Please refer to the Administrator Guide for more information on supported features.

Do you wish to continue?

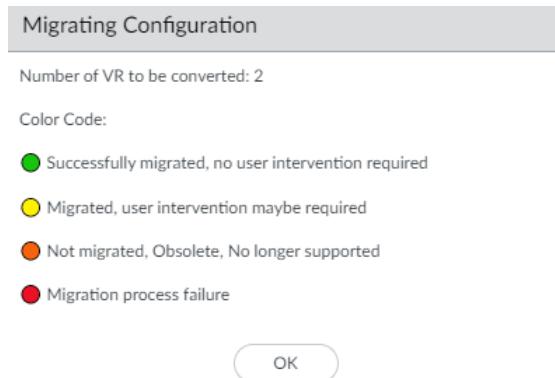
Yes **Skip** **Cancel**

Select **Yes** to have the migration script convert each virtual router to a logical router and migrate your configuration to the advanced routing engine. (Select **Skip** to restart

Advanced Routing

the system with an empty configuration. Select **Cancel** to cancel the process to enable Advanced Routing.)

7. Click **OK** to approve the migration.



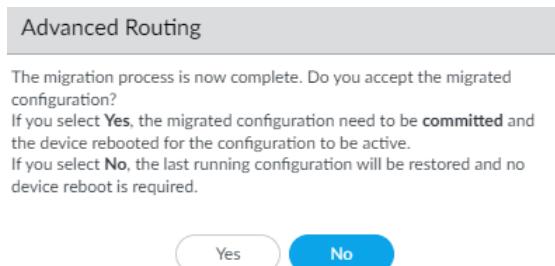
8. The virtual routers, links to the logical routers, and their color-coded status are listed. Resolve any issues that require user intervention. Select **Continue**

Virtual Router		
Migration		
NAME	INTERNAL LINK	STATUS
VR-North	Open in Network -> Logical Routers	●
VR-Tunnel-North	Open in Network -> Logical Routers	●

Legend: ● Successful ● User Intervention ● Obsolete / Not Supported ● Failed

Continue

9. Click **Yes** to accept the migrated configuration.



10. (PAN-OS 10.2.5 and later 10.2 releases) Click **OK**.

Advanced Routing Changed

Advanced Routing is enabled.
This window will be reloaded. However, Commit and Reboot are still required to make the new config effective.

OK

11. Commit and then select **Device > Setup > Operations** and **Reboot Device**. Then log back into the firewall.

If the migration is not successful, generate the technical support file, log in to [Palo Alto Networks Customer Support Portal](#), and report your issues to get help with your product.

12. **(Optional)** After successful migration, you can delete all virtual routers using the configuration mode CLI command:

1. [Access the CLI](#).

2. Execute the following command to remove all configurations from the legacy routing engine:

```
username@hostname# delete network virtual-router <vr-name>
```



You can delete virtual routers if you are going to make changes in the logical router configuration, which makes the virtual router configuration obsolete, causing commit failures. Although deleting virtual routers will avoid commit failures, be aware that deleting virtual routers will also permanently remove all configuration from the legacy routing engine and you won't be able to get the configuration back.

3. Commit the changes to the firewall.

```
username@hostname# commit
```



When configuring in Panorama, you can select **Network > Virtual Routers** to delete all virtual routers. Commit the changes and push them to the relevant firewalls before continuing.

STEP 3 | Log back into the firewall.

STEP 4 | Select Network.

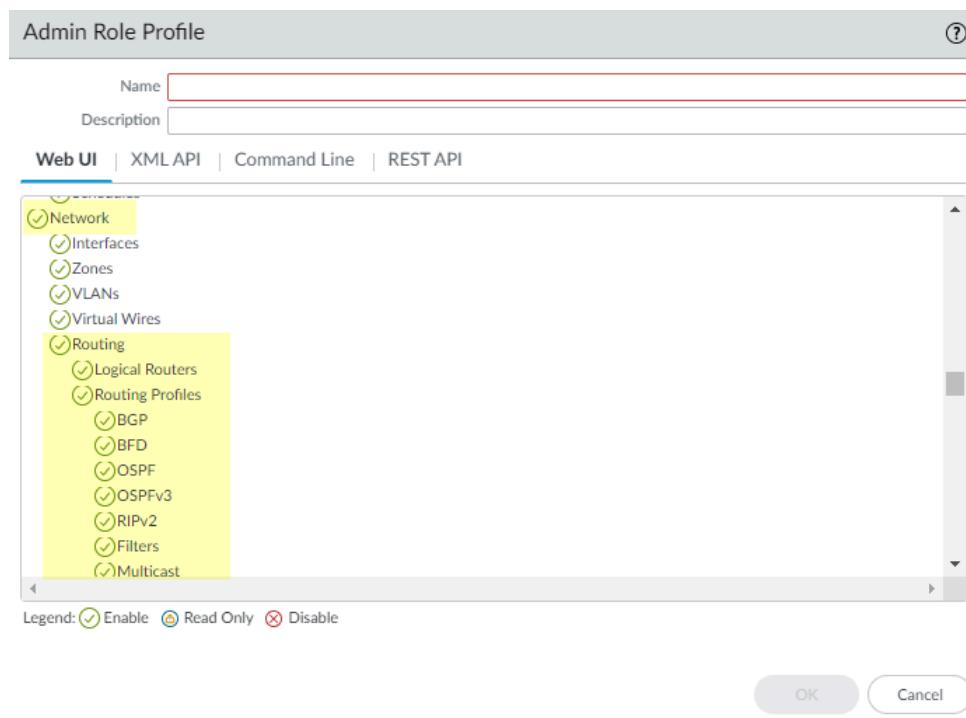
Notice the menu items, which are more industry-standard and more detailed than the single item (Virtual Routers) on the legacy menu. **Routing** includes **Logical Routers** and **Routing Profiles**, which include **BGP**, **BFD**, **OSPF**, **OSPFv3**, **RIPV2**, **Filters**, and **Multicast**.

NAME	INTERFACES	GENERAL	BGP	STATIC	RUNTIME STATS
LR-1	ethernet1/1 ethernet1/4.1 loopback.1	ECMP Max Paths: 2	Enabled Peer Group Count: 1 Peer Count: 1		More Runtime Stats
LR-3	ethernet1/3 ethernet1/4.3	ECMP Max Paths: 2	Peer Group Count: 0		More Runtime Stats

STEP 5 | Select **Interfaces** and configure one or more [Layer 3 interfaces](#) with a static IP address or [Configure an Interface as a DHCP Client](#).

STEP 6 | **(Optional)** Create an Admin Role Profile to control granular access to logical routers and routing profiles for an Advanced Routing Engine.

1. Select **Device > Admin Roles** and Add an Admin Role Profile by **Name**.
2. Select **Web UI**.
3. **Enable, Disable**, or select **Read Only** the following options: **Network, Routing, Logical Routers, Routing Profiles, BGP, BFD, OSPF, OSPFv3, RIPv2, Filters, and Multicast** (default is Enable).



4. Click **OK**.
5. Assign the role to an administrator. [Configure a Firewall Administrator Account](#).

STEP 7 | Commit the changes.

STEP 8 | Continue by [configuring a logical router](#).

If you downgrade from PAN-OS 10.2.5 or 10.2.4-h2 to a previous version, you must remove the SD-WAN virtual interface (VIF) from the logical router configurations before attempting a downgrade procedure.

That is, you must select a different interface instead of SD-WAN VIF interface in the following **Logical Router** configurations:

- Select **Logical Router > General > Interface** and specify a different **Interface**.
- Select **Logical Router > Static** and specify a different **Interface**.
- Select **Logical Router > BGP > Peer Group > Peer** and specify a different **Interface** for **Local Address**.

Logical Router Overview

The firewall uses logical routers to obtain Layer 3 routes to other subnets by you manually defining static routes or through participation in one or more Layer 3 routing protocols (dynamic routes). The routes that the firewall obtains through these methods populate the IP routing information base (RIB) on the firewall. When a packet is destined for a different subnet than the one it arrived on, the logical router obtains the best route from the RIB, places it in the forwarding information base (FIB), and forwards the packet to the next hop router defined in the FIB. The firewall uses Ethernet switching to reach other devices on the same IP subnet. (An exception to one best route going in the FIB occurs if you are using [ECMP](#), in which case all equal-cost routes go in the FIB.)

The Ethernet, VLAN, and tunnel interfaces defined on the firewall receive and forward Layer 3 packets. The destination zone is derived from the outgoing interface based on the forwarding criteria, and the firewall consults policy rules to identify the security policies that it applies to each packet. In addition to routing to other network devices, logical routers can route to other logical routers within the same firewall if a next hop is specified to point to another logical router.

You can [Configure Layer 3 Interfaces](#) to participate with dynamic routing protocols (BGP, OSPF, OSPFv3, or RIP) as well as add static routes. You can also create multiple logical routers, each maintaining a separate set of routes that aren't shared between logical routers, enabling you to configure different routing behaviors for different interfaces.

You can configure dynamic routing from one logical router to another by configuring a loopback interface in each logical router, creating a static route between the two loopback interfaces, and then configuring a dynamic routing protocol to peer between these two interfaces. The firewall supports only one hop between logical routers. For example, with logical routers A, B, and C, a route cannot go from A to B to C; it would have to go from A to C.

Each Layer 3 Ethernet, loopback, VLAN, and tunnel interface defined on the firewall must be associated with a logical router. While each interface can belong to only one logical router, you can configure multiple routing protocols and static routes for a logical router. Regardless of the static routes and dynamic routing protocols you configure for a logical router, one general configuration is required.

Configure a Logical Router

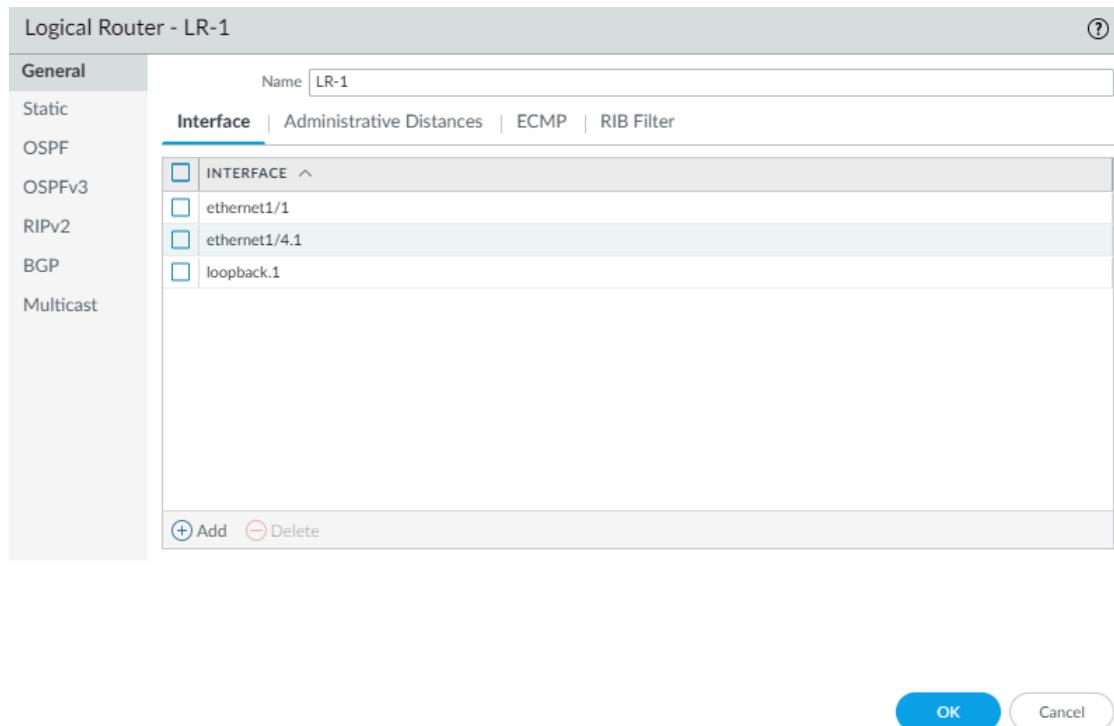
In order to perform network routing, the Advanced Routing Engine requires you to configure at least one [logical router](#); there is no default logical router. A logical router maintains a separate routing information base and keeps routes from exposure to other logical routers. The [number of logical routers supported](#) for an Advanced Routing Engine varies based on firewall model.

Before you can configure a logical router, you must [Enable Advanced Routing](#).

STEP 1 | Select **Network > Routing > Logical Routers** and **Add** a logical router by **Name** using a maximum of 31 characters. The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore (_) or hyphen(-). No dot (.) or space is allowed.

STEP 2 | Add interfaces to the logical router.

1. While still on the Logical Router **General** tab, select the **Interface** tab.
2. **Add** an interface to the logical router by selecting from the list of interfaces. An interface can belong to only one logical router. Repeat to add more interfaces, as in the following example for the logical router named LR-1:



STEP 3 | (Optional) Select **Administrative Distances** to change the global administrative distance (from the default setting) for various types of routes.

The screenshot shows the 'Logical Router - LR-1' configuration window. The 'Name' field is set to 'LR-1'. The 'Administrative Distances' tab is selected. A sidebar on the left lists route types: General, Static, OSPF, OSPFv3, RIPv2, BGP, and Multicast. The main area displays administrative distances for each type:

Type	Administrative Distance
Static	10
Static IPv6	10
OSPF Intra Area	110
OSPF Inter Area	110
OSPF External	110
OSPFv3 Intra Area	110
OSPFv3 Inter Area	110
OSPFv3 External	110
BGP AS Internal	200
BGP AS External	20
BGP Local Route	20
RIP	120

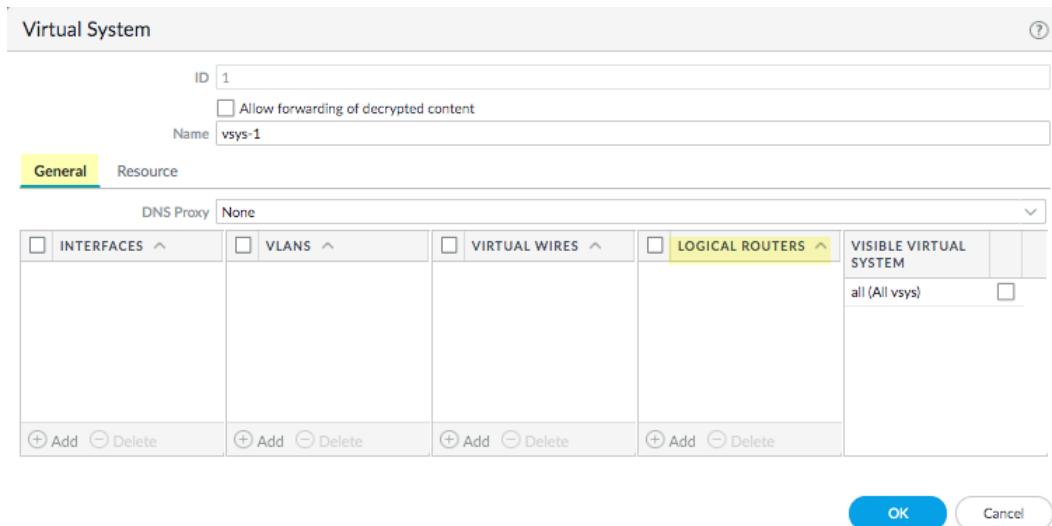
At the bottom right are 'OK' and 'Cancel' buttons.

- **Static**—Range is 1 to 255; default is 10.
- **Static IPv6**—Range is 1 to 255; default is 10.
- **OSPF Intra Area**—Range is 1 to 255; default is 110.
- **OSPF Inter Area**—Range is 1 to 255; default is 110.
- **OSPF External**—Range is 1 to 255; default is 110.
- **OSPFv3 Intra Area**—Range is 1 to 255; default is 110.
- **OSPFv3 Inter Area**—Range is 1 to 255; default is 110.
- **OSPFv3 External**—Range is 1 to 255; default is 110.
- **BGP AS Internal**—Range is 1 to 255; default is 200.
- **BGP AS External**—Range is 1 to 255; default is 20.
- **BGP Local Route**—Range is 1 to 255; default is 20.
- **RIP**—Range is 1 to 255; default is 120.

STEP 4 | Click **OK**.

STEP 5 | (On a firewall supporting multiple virtual systems) Assign the logical routers to a virtual system.

1. Select **Device > Virtual Systems** and select a virtual system and **General**.
2. Add one or more **Logical Routers**.
3. Click **OK**.



STEP 6 | Click **OK**.

STEP 7 | (Optional) Configure ECMP for a logical router by navigating to **Network > Routing > Logical Routers**, selecting a logical router, and then **General > ECMP**. Configure ECMP for a logical router much as you would for a virtual router on a legacy routing engine.

 *ECMP is not supported for equal-cost routes where one or more of those routes has a virtual router or logical router as the next hop. None of the equal-cost routes will be installed in the Forwarding Information Base (FIB).*

STEP 8 | Commit the changes.

STEP 9 | For a firewall with a pre-existing configuration, select **Device > Setup > Operations** and **Reboot Device**. Then log back into the firewall.

STEP 10 | (Optional) View Runtime Stats for a logical router.

1. Select **Network > Routing > Logical Routers** and for a specific logical router, select **More Runtime Stats** on the far right.
2. To see the route tables for all protocols, on the **Routing** tab, select **Route Table** and **Display Address Family: IPv4 and IPv6, IPv4 Only, or IPv6 Only**.

The screenshot shows the PAN-OS configuration interface for a logical router named 'LR-1'. At the top, there's a navigation bar with tabs for 'Routing', 'OSPF', 'OSPFv3', 'BGP', 'Multicast', and 'BFD Summary Information'. Below this, a sub-navigation bar has 'Route Table' selected. A search bar at the top of the main content area contains a magnifying glass icon and the text '0 items'. Below the search bar is a table header with columns: DESTINATION, NEXT HOP, PROTOCOL, METRIC, SELECTED, AGE, ACTIVE, and INTERFACE. The table body is completely empty. At the bottom left is a 'Refresh' button, and at the bottom right is a 'Close' button.

3. To see entries in the Forwarding Information Base (FIB), select **Forwarding Table**.
4. Select **Static Route Monitoring** to see the static routes you are monitoring.
5. Select the **BGP** tab and then **Summary** to see BGP settings.
6. Select **Peer** to see BGP peer settings.
7. Select **Peer Group** to see BGP peer group settings.
8. Select **Route** and **Display Address Family: IPv4 and IPv6, IPv4 Only, or IPv6 Only** to see the attributes of BGP routes.

STEP 11 | Access the CLI to view advanced routing information. The PAN-OS CLI Quick Start lists the commands in the [CLI Cheat Sheet: Networking](#).

Create a Static Route

Create a static route for a logical router on an [Advanced Routing Engine](#).

STEP 1 | Configure a Logical Router.

STEP 2 | Create a static route.

1. Select **Network > Routing > Logical Routers** and select the logical router.
2. Select **Static** and **Add an IPv4 or IPv6** static route by **Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. For **Destination**, enter the route and netmask (for example, 192.168.2.0/24 for an IPv4 address or 2001:db8:123:1::0/64 for an IPv6 address). If you're creating a default route, enter the default route (0.0.0.0/0 for an IPv4 address or ::/0 for an IPv6 address). Alternatively, you can select or create an address object of type IP Netmask.
4. For **Interface**, specify the outgoing interface for packets to use to go to the next hop. Specifying an interface provides stricter control over which interface the firewall uses rather than using the interface in the route table for the next hop of this static route.
5. For **Next Hop**, select one of the following:
 - **IP Address or IPv6 Address**—Enter the IP address (for example, 192.168.56.1 or 2001:db8:49e:1::1) when you want to route to a specific next hop. You must **Enable IPv6 on the interface** (when you [Configure Layer 3 Interfaces](#)) to use an IPv6 next hop address. If you're creating a default route, for **Next Hop** you must select **IP Address** and enter the IP address for your internet gateway (for example, 192.168.56.1 or 2001:db8:49e:1::1). Alternatively, you can create an address object

of type IP Netmask. The address object must have a netmask of /32 for IPv4 or /128 for IPv6.

- **Next LR**—Select to make the next logical router (in the list of logical routers) the next hop.
 - **FQDN**—Enter a Fully Qualified Domain Name.
 - **Discard**—Select to drop packets that are addressed to this destination.
 - **None**—Select if there is no next hop for the route. For example, a point-to-point connection does not require a next hop because there is only one way for packets to go.
6. Enter the **Admin Dist** for the static route (range is 10 to 240; default is 10). This value overrides the **Static** or **Static IPv6** administrative distance specified for the logical router.
 7. Enter a **Metric** for the static route (range is 1 to 65,535; default is 10).
 8. (**Optional**) If you want to use BFD, select a **BFD Profile** you created, or select the **default** profile, or [create a BFD profile](#) to apply to the static route; default is **None (Disable BFD)**.

The screenshot shows the 'Static Routes - IP' configuration dialog. At the top, there are fields for 'Name', 'Destination', 'Interface' (set to 'None'), 'Next Hop' (set to 'None'), 'Admin Dist' (set to '[10 - 240]'), 'Metric' (set to '10'), and 'BFD Profile' (set to 'None'). Below these, the 'Path Monitoring' section is expanded, showing a checkbox for 'Enable', a 'Failure Condition' radio button set to 'Any' (with 'All' also available), and a 'Preemptive Hold Time (min)' input field set to '2'. A table below lists monitored static routes with columns: NAME, ENABLE, SOURCE IP, DESTINATION IP, PING INTERVAL(SEC), and PING COUNT. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

STEP 3 | (**Optional**) Configure path monitoring for the static route; you can monitor up to 128 static routes.

1. Select **Path Monitoring** to allow configuration of path monitoring (default is disabled).
2. **Enable** path monitoring (default is disabled).
3. **Failure Condition** determines whether path monitoring for the static route is based on one (any) or all monitored destinations. Select whether **Any** or **All** of the monitored destinations for the static route must be unreachable by ICMP for the firewall to remove

the static route from the RIB and FIB and add the static route that has the next lowest metric (going to the same destination) to the FiB.



Select **All** to avoid the possibility of any single monitored destination signaling a route failure when the destination is simply offline for maintenance, for example.

4. **(Optional)** Specify the **Preemptive Hold Time (min)**, the number of minutes a downed path monitor must remain in Up state before the firewall reinstalls the static route into the RIB; range is 0 to 1,440; default is 2. A setting of 0 (zero) causes the firewall to reinstall the route into the RIB immediately upon the path monitor coming up.

The path monitor evaluates all of its monitored destinations for the static route and comes up based on the **Any** or **All** failure condition. If a link goes down or flaps during the hold time, when the link comes back up, the path monitor resumes and the Preemptive Hold Time is reset, causing the timer to restart from zero.

5. **Add** a path monitoring destination by **Name**.

The dialog box is titled "Path Monitoring Destination". It contains the following fields:

- Name:** A text input field with the placeholder "Name" and a cursor.
- Enable:** A checked checkbox labeled "Enable".
- Source IP:** A dropdown menu showing a list of available source IP addresses.
- Destination IP:** A dropdown menu showing a list of available destination IP addresses.
- Ping Interval(sec):** A numeric input field set to "3".
- Ping Count:** A numeric input field set to "5".

At the bottom right are two buttons: "OK" and "Cancel".

6. **Enable** the path monitoring destination.
7. For **Source IP**, select the IP address that the firewall uses in the ICMP ping to the monitored destination:
 - If an interface has multiple IP addresses, select one.
 - If you select an interface, the firewall uses the first IP address assigned to the interface by default.
 - If you select **DHCP (Use DHCP Client address)**, the firewall uses the address that DHCP assigned to the interface. To see the DHCP address, select **Network >**

Interfaces > Ethernet and in the row for the Ethernet interface, click on **Dynamic DHCP Client**. The IP Address displays in the Dynamic IP Interface Status window.

8. For **Destination IP**, enter an IP address or address object to which the firewall will monitor the path. The monitored destination and static route destination must use the same address family (IPv4 or IPv6).



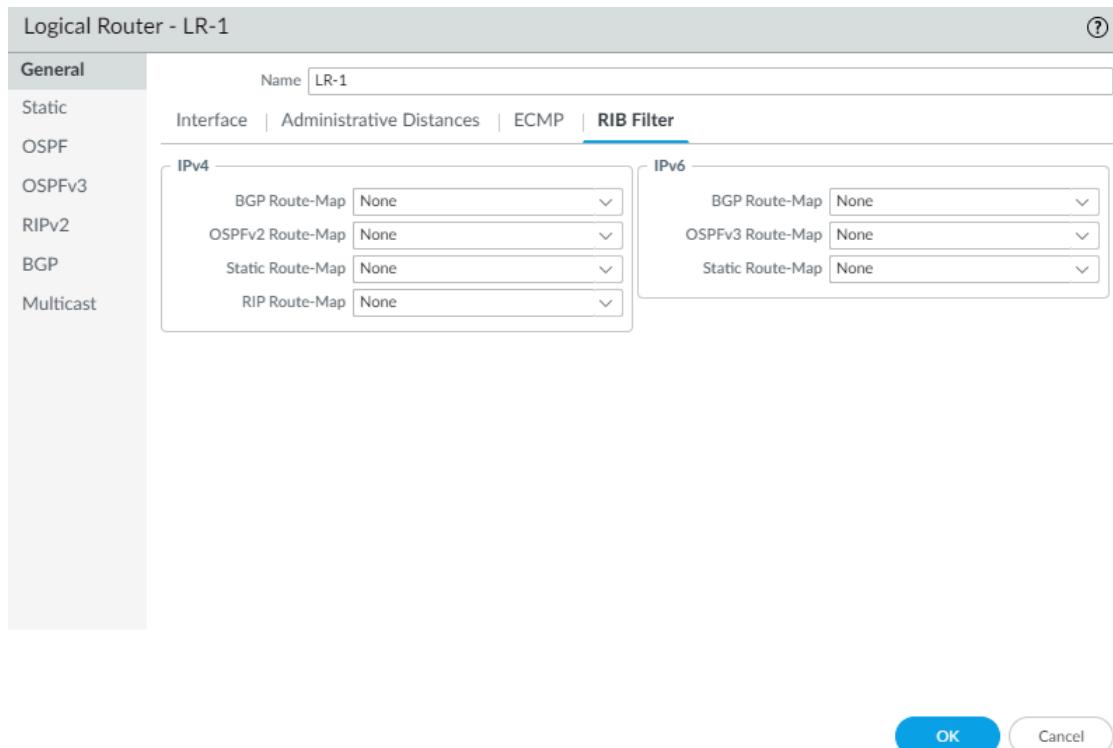
The destination IP address should belong to a reliable endpoint; you shouldn't base path monitoring on a device that itself is unstable or unreliable.

9. **(Optional)** Specify the ICMP Ping Interval (sec) in seconds to determine how frequently the firewall monitors the path (range is 1 to 60; default is 3).
10. **(Optional)** Specify the ICMP Ping Count of packets that don't return from the destination before the firewall considers the static route down and removes it from the RIB and FIB (range is 3 to 10; default is 5).
11. Click **OK** to save the path monitor destination.
12. Click **OK** twice to save the static route.

STEP 4 | (Optional) Control the static routes that are placed in the global RIB.

You might configure static routes and redistribute them, but not want them in the protocol's local route table or global RIB. You might want to add only specific static routes to the global RIB.

1. Select **Network > Routing > Logical Routers** and select a logical router.
2. Select **RIB Filter** to allow routes into or prevent routes from being added to the global RIB.



3. To filter IPv4 static routes and connected routes, for **Static Route-Map**, select a Redistribution Route Map or [create a new one](#).
4. To filter IPv6 static routes and connected routes, for **Static Route-Map**, select a Redistribution Route Map or [create a new one](#).
5. Click **OK**.

STEP 5 | (Optional) Change the default administrative distances for static IPv4 and static IPv6 routes within a logical router.

STEP 6 | Commit the changes.

STEP 7 | Access the CLI to view the static route path monitor: **show advanced-routing static-route-path-monitor**. The PAN-OS CLI Quick Start lists additional commands in the [CLI Cheat Sheet: Networking](#).

Configure BGP on an Advanced Routing Engine

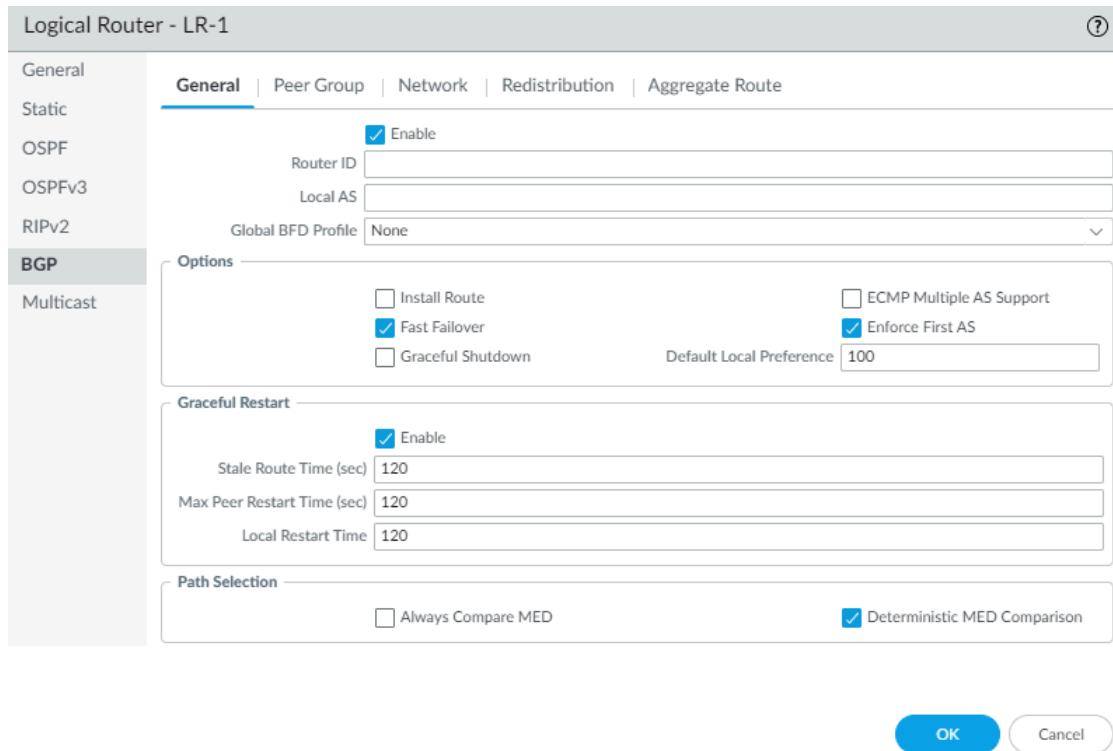
Perform the following task to configure BGP for a logical router on an [Advanced Routing Engine](#).

Before you configure BGP, consider the many useful [routing profiles](#) and [filters](#) that you can apply to BGP peer groups, peers, redistribution rules, and aggregate route policies, and thereby save configuration time and maintain consistency. You can create profiles and filters in advance or as you progress through configuring BGP.

STEP 1 | Configure a Logical Router.

STEP 2 | Enable BGP and configure general BGP settings.

1. Select **Network > Routing > Logical Routers** and select a logical router.
2. Select **BGP > General** and **Enable BGP** for this logical router.



3. Assign a **Router ID** to BGP for the logical router, which is typically an IPv4 address to ensure the Router ID is unique.
4. Assign the **Local AS**, which is the number of the AS to which the logical router belongs; range is 1 to 4,294,967,295.
5. If you want to apply BFD to BGP, for **Global BFD Profile** select a BFD profile you created, or select the **default** profile, or [create a new BFD profile](#); default is **None (Disable BFD)**.
6. Select **Install Route** to install learned BGP routes into the global routing table; default is disabled.
7. Select **Fast Failover** to have BGP terminate a session with an adjacent peer if the link to that peer goes down, without waiting for the [Hold Time](#) to expire; default is enabled.
8. Select **Graceful Shutdown** to have BGP lower the preference of eBGP peering links during a maintenance operation so that BGP can choose and propagate alternative paths, based on [RFC 8326](#); default is disabled.
9. Select **ECMP Multiple AS Support** if you configured ECMP and you want to run ECMP over multiple BGP autonomous systems; default is disabled.
10. **Enforce First AS** to cause the firewall to drop an incoming Update packet from an eBGP peer that does not list the eBGP peer's own AS number as the first AS number in the [AS_PATH](#) attribute; default is enabled.
11. Specify the **Default Local Preference** that can be used to determine preferences among different paths; range is 0 to 4,294,967,295; default is 100.

12. Enable Graceful Restart and configure the following timers:

- **Stale Route Time (sec)**—Specifies the length of time, in seconds, that a route can stay in the stale state (range is 1 to 3,600; default is 120).
- **Max Peer Restart Time (sec)**—Specifies the maximum length of time, in seconds, that the local device accepts as a grace period restart time for peer devices (range is 1 to 3,600; default is 120).
- **Local Restart Time**—Specifies the length of time, in seconds, that the local device waits to restart. This value is advertised to peers (range is 1 to 3,600; default is 120).

13. For Path Selection:

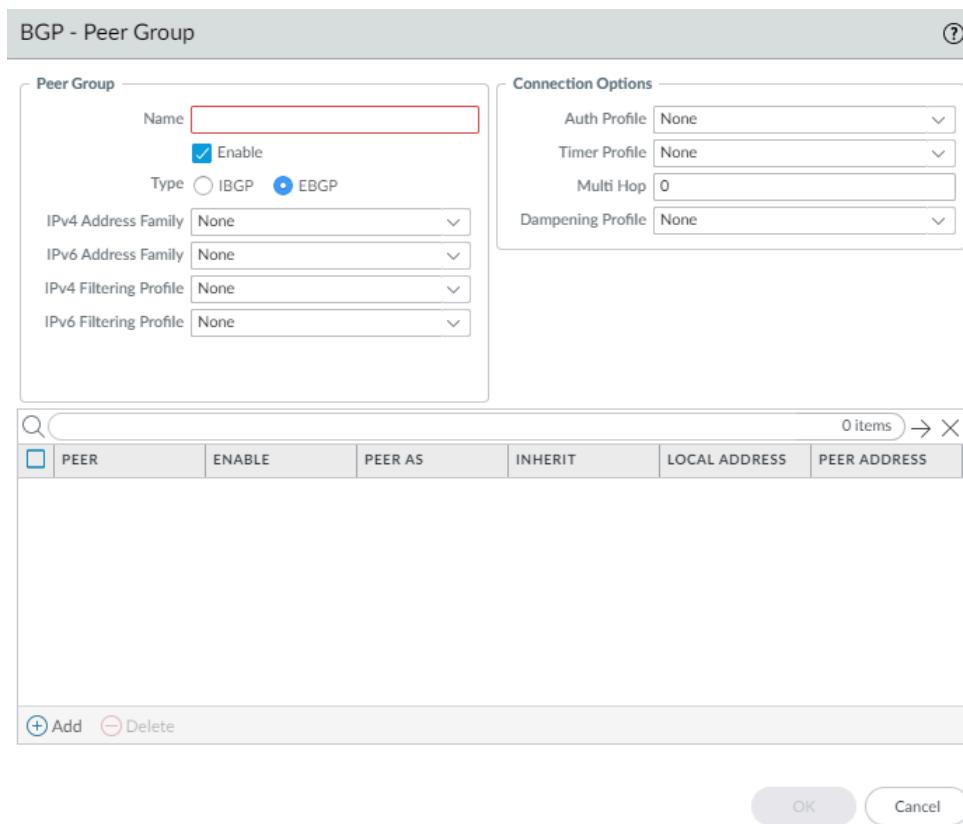
- **Always Compare MED**—Enable this comparison to choose paths from neighbors in different autonomous systems; default is disabled.
- **Deterministic MED Comparison**—Enable this comparison to choose between routes that are advertised by IBGP peers (BGP peers in the same autonomous system); default is enabled.

14. Click **OK**.

STEP 3 | Configure a BGP peer group.

1. Select > **Routing** > **Logical Routers** and select a logical router.
2. Select **BGP** > **Peer Group** and **Add** a BGP peer group by **Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and contain zero or more alphanumeric characters, underscore, or hyphen.

No dot (.) or space is allowed. The name must be unique within the logical router and across all logical routers.



3. **Enable** the peer group.
4. Select the **Type** of peer group: **IBGP** or **EBGP**.
5. To specify many **IPv4 Address Family** options for the peer group, select an **AFI Profile** that you created, select the **default** profile, or [create a new BGP Address Family profile](#); the default is **None**.
6. To specify many **IPv6 Address Family** options for the peer group, select an **AFI Profile** that you created, select the **default** profile, or [create a new BGP Address Family profile](#); the default is **None**.
7. To apply **IPv4 Filtering Profile** options to the peer group, select a **BGP Filtering Profile** that you created or [create a new BGP Filtering profile](#); the default is **None**.



A **BGP Filtering Profile** describes how to configure many BGP options for IPv4, such as import or export BGP routes, accept or prevent routes being added to the local BGP RIB, conditionally advertise routes, and unsuppress dampened or summarized routes.

8. To apply **IPv6 Filtering Profile** options to the peer group, select a **BGP Filtering Profile** that you created or [create a new BGP Filtering profile](#); the default is **None**.



A **BGP Filtering Profile** describes how to configure many BGP options for IPv6, such as import or export BGP routes, accept or prevent routes being added to the local BGP RIB, conditionally advertise routes, and unsuppress dampened or summarized routes.

9. For Connection Options, select an **Auth Profile** or [create a new BGP Authentication profile](#) to control MD5 authentication between BGP peers in the peer group. Default is **None**.
10. Select a **Timer Profile** or [create a new BGP Timer Profile](#) to control various BGP timers that affect keepalive and update messages that advertise routes. Default is **None**.
11. Set **Multi Hop**—the time-to-live (TTL) value in the IP header (range is 0 to 255; default is 0). The default value of 0 means 1 for eBGP. The default value of 0 means 255 for iBGP.
12. Select a **Dampening Profile** or [create a new Dampening Profile](#) to determine how to penalize a flapping route to suppress it from being used until it stabilizes. Default is **None**.

STEP 4 | Add a BGP peer to the peer group.

1. **Add a peer by Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and contain zero or more

alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed. The name must be unique within the logical router and across all logical routers.

2. **Enable** the peer; default is enabled.
3. Select **Passive** to prevent the peer from initiating a session with its neighbors; default is disabled.
4. Enter the **Peer AS** to which the peer belongs; range is 1 to 4,294,967,295.
5. Select **Addressing** and select whether the peer will **Inherit** IPv4 and IPv6 AFI and filtering profiles from the peer group: **Yes** (default) or **No**.
6. If you chose **Yes**, specify the following for the peer:
 - For **Local Address**, select the **Interface** for which you are configuring BGP. If the interface has more than one IP address, select the **IP Address** for that interface to be the BGP peer.
 - For **Peer Address**, select either **IP** and select the IP address or select or create an address object, or select **FQDN** and enter the FQDN or address object that is type FQDN.



The firewall uses only one IP address (from each IPv4 or IPv6 address type) from the DNS resolution of the FQDN. If the DNS resolution returns more than one address, the firewall uses the preferred IP address that matches the IP family type (IPv4 or IPv6) configured for the BGP peer. The preferred IP address is the first address the DNS server returns in its initial response. The firewall retains this address as preferred as long as the address appears in subsequent responses regardless of its order.

BGP - Peer Group - Peer

Name

Enable
 Passive

Peer AS

[Addressing](#) | [Connection Options](#) | [Advanced](#)

Inherit Yes No

Local Address

Interface
IP Address

Peer Address

IP

[OK](#) [Cancel](#)

7. If you chose **No** for **Inherit** addressing from the peer group, specify the following for the peer:

- To specify many **IPv4 Address Family** options for the peer, select an **AFI Profile** that you created, select the **default** profile, select **inherit (Inherit from Peer-Group)**, or [create a new BGP Address Family profile](#); the default is **none (Disable IPv4 AFI)**.



The AFI Profile allows you to specify that the peer is a Route Reflector client. The Route Reflector reflects all the advertisements from all its peers to all the other peers, thus avoiding the need for the iBGP to be fully meshed. If you declare the peer a Route Reflector client, the BGP process reflects all of the updates to that peer.

- To specify many **IPv6 Address Family** options for the peer, select an **AFI Profile** that you created, select **inherit (Inherit from Peer-Group)**, or [create a new BGP Address Family profile](#); the default is **none (Disable IPv6 AFI)**.



The AFI Profile allows you to specify that the peer is a Route Reflector client. The Route Reflector reflects all the advertisements from all its peers to all the other peers, thus avoiding the need for the iBGP to be fully meshed. If you declare the peer a Route Reflector client, the BGP process reflects all of the updates to that peer.

- To apply **IPv4 Filtering Profile** options to the peer, select a **BGP Filtering Profile** that you created, select **inherit (Inherit from Peer-Group)**, or [create a new BGP Filtering profile](#); the default is **none (Disable IPv4 Filtering)**.
- To apply **IPv6 Filtering Profile** options to the peer, select a **BGP Filtering Profile** that you created, select **inherit (Inherit from Peer-Group)**, or [create a new BGP Filtering profile](#); the default is **none (Disable IPv6 Filtering)**.
- For **Local Address**, select the **Interface** for which you are configuring BGP. If the interface has more than one IP Address, select the **IP address** for that interface to be the BGP peer.
- For **Peer Address**, select either **IP** and select the IP address or select or create an address object, or select **FQDN** and enter the FQDN or address object that is type FQDN.

BGP - Peer Group - Peer

Name

Enable
 Passive

Peer AS

Addressing | Connection Options | Advanced

Inherit Yes No

IPv4 Address Family	none
IPv6 Address Family	none
IPv4 Filtering Profile	none
IPv6 Filtering Profile	none

Local Address

Interface	<input type="text"/>
IP Address	None

Peer Address

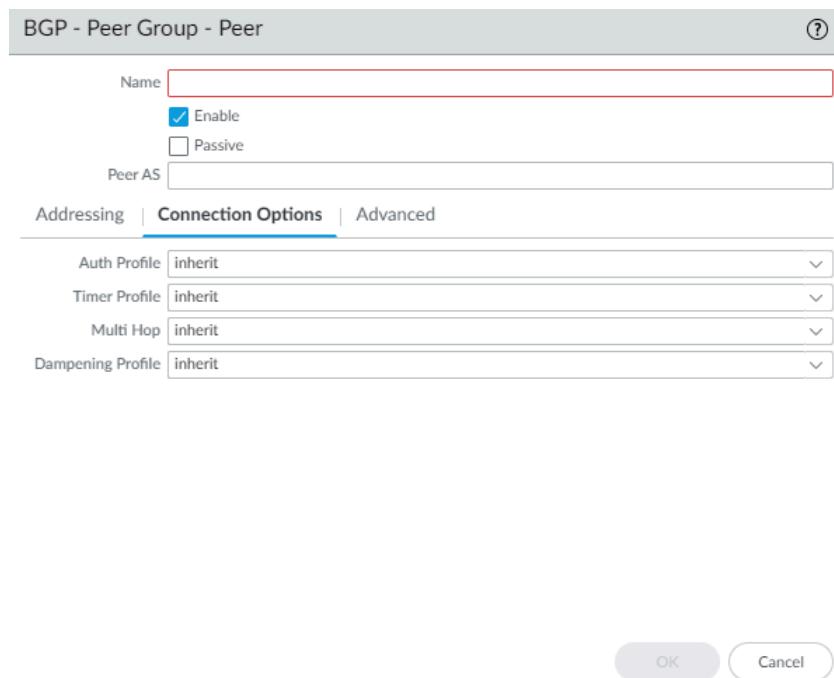
Type	IP	<input type="text"/>
------	----	----------------------

OK Cancel



If a BGP peer group has both an IPv4 Address Family profile and an IPv6 Address Family profile applied to it, all peers belonging to that peer group will automatically have Addressing set to Inherit No (because a peer cannot use both address families). All peers in the peer group will also have IPv4 Address Family profile, IPv6 Address Family profile, IPv4 Filtering Profile, and IPv6 Filtering Profile set to **none** by default. You can select inherit (Inherit from Peer-Group) or override the peer group by selecting a specific profile for the peer. For example, you can configure a peer to **inherit** the IPv4 Address Family profile and **inherit** the IPv4 Filtering Profile, and select an IPv6 Address Family profile and IPv6 Filtering Profile to override those profiles from the peer group.

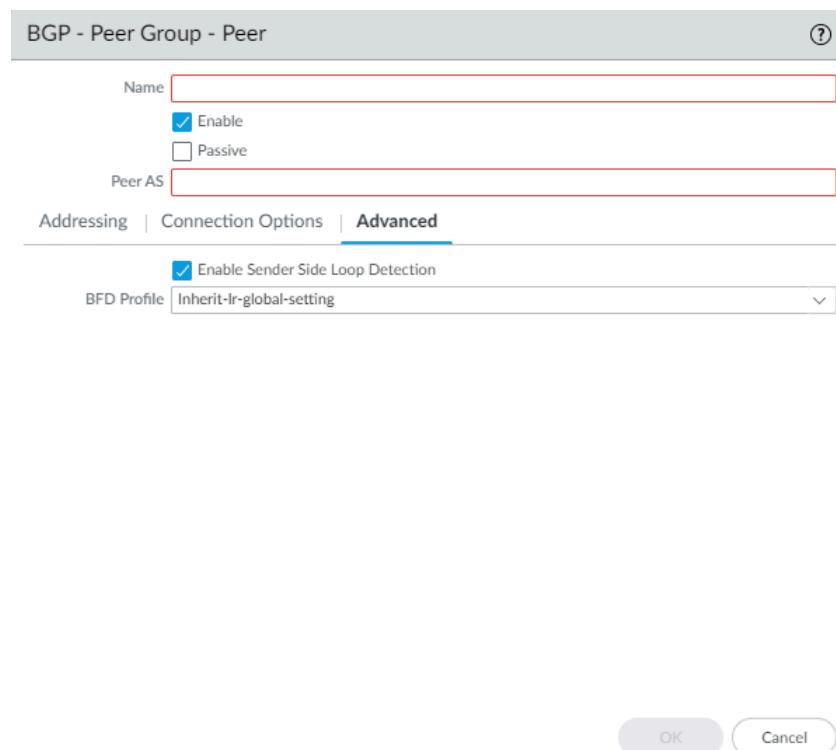
8. Select **Connection Options** for the peer in order to apply settings that differ from those of the peer group.



9. Select an **Auth Profile**, **inherit (Inherit from Peer-Group)** (the default), or [create a new BGP Authentication profile](#) to control MD5 authentication between BGP peers.
10. Select a **Timer Profile**, **inherit (Inherit from Peer-Group)** (the default), [create a new BGP Timer Profile](#), or select the **default** profile to control various BGP timers that affect keepalive and update messages that advertise routes.
11. Set **Multi Hop**, which is the time-to-live (TTL) value in the IP header (range is 0 to 255). The default setting is **inherit (Inherit from Peer-Group)**.
12. Select a **Dampening Profile**, **inherit (Inherit from Peer-Group)** (the default), or [create a new Dampening Profile](#) to determine how to penalize a flapping route to suppress it from being used until it stabilizes.
13. Select **Advanced** and **Enable Sender Side Loop Detection** to have the firewall check the AS_PATH attribute of a route in the BGP RIB before it sends the route in an update, to ensure that the peer AS number isn't in the AS_PATH list. The firewall doesn't advertise the route if the peer AS number is in the AS_PATH list. Usually, the receiver detects

loops, but this optimization feature has the sender perform the loop detection. Disable this feature to have the receiver perform loop detection.

14. To apply a **BFD Profile** to the peer (which overrides the BFD setting for BGP, as long as BFD is not disabled for BGP at the logical router level), select one of the following:
 - The **default** profile.
 - An existing BFD profile.
 - **Inherit-Ir-global-setting (Inherit Protocol's Global BFD Profile)** (default)—Peer inherits the BFD profile that you selected globally for BGP for the logical router.
 - **None (Disable BFD)** for the peer.
 - [Create a new BFD profile](#).



15. Click **OK**.

STEP 5 | Specify network prefixes to advertise to neighbors.

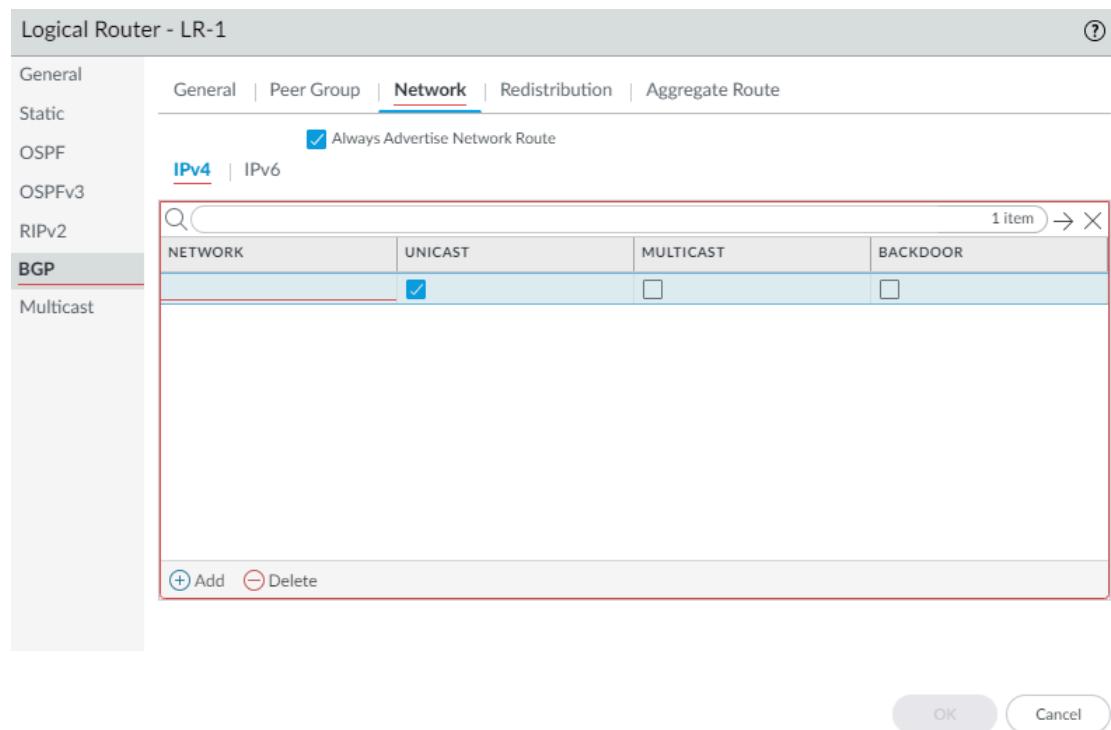


The **Network** functionality is especially helpful after moving a firewall to a different subnet or after a temporary change in a network.

1. Select **Network**.
2. **Always Advertise Network Route** (default is enabled) to always advertise the configured network routes to BGP peers, regardless of whether they are reachable or not. If this is

unchecked, the firewall advertises the network routes only if they are resolved using the local route table.

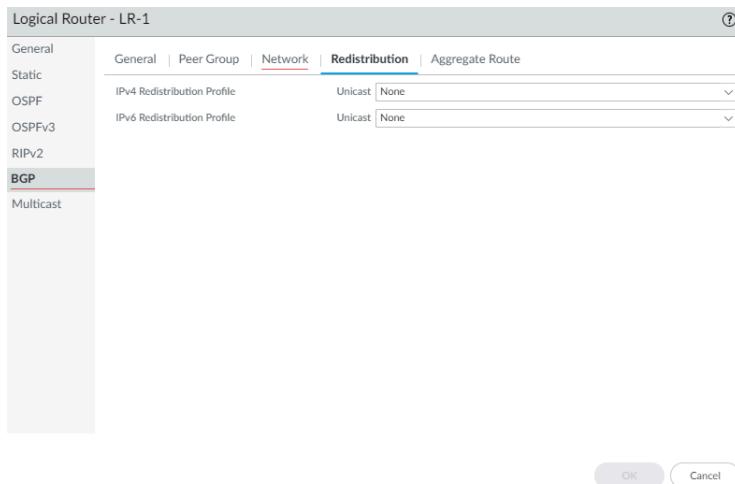
3. Select **IPv4** or **IPv6** to select type of prefix.
4. Add a **Network** prefix to advertise to neighbors.
5. Select **Unicast** to advertise this network route in the Unicast Address Family; default is enabled. If unchecked, the firewall does not advertise the route in the Unicast SAFI.
6. (**IPv4 only**) Select **Multicast** to advertise this network route into the Multicast Address Family. Default is disabled; the firewall does not advertise this network route in the Multicast SAFI.
7. (**IPv4 only**) Select **Backdoor** to prevent BGP from advertising the prefix outside of the AS and instead to keep the route within the AS. A backdoor is a BGP route that has a higher administrative distance than an IGP route. Internally, the administrative distance for the prefix is increased so that the prefix isn't preferred, but is still available if needed, in the event of a link failure elsewhere. Default is disabled.



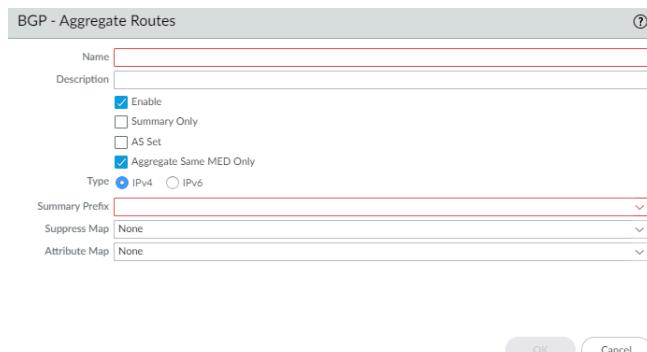
STEP 6 | Redistribute static, connected, OSPF, OSPFv3, or RIPv2 routes into BGP.

 Within a BGP Redistribution Profile, use the flexibility of route maps to specify conditions that determine which routes to redistribute and to specify which attributes to set.

1. Select **Redistribution**.
2. To redistribute IPv4 routes, for **IPv4 Redistribution Profile -- Unicast**, select a [BGP Redistribution profile](#) or create a new Redistribution profile; default is **None**.
3. To redistribute IPv6 routes, for **IPv6 Redistribution Profile -- Unicast**, select a [BGP Redistribution profile](#) or create a new Redistribution profile; default is **None**.

**STEP 7 |** Create an aggregate route policy to summarize routes that BGP learns and then advertises to peers.

1. Select **Aggregate Route** and Add an aggregate route policy by **Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain alphanumeric characters, underscores, and hyphens. No dot (.) or space is allowed.
2. Enter a helpful **Description** of the policy.
3. **Enable** the policy.



4. Select **Summary Only** to advertise to neighbors only the **Summary Prefix** and not the routes that were summarized; this reduces traffic and avoids increasing the size of the neighbors' routing tables unnecessarily (default is disabled). If you want to advertise both

the aggregate route and the individual routes that make up the aggregate route, leave **Summary Only** unchecked.



Summary Only and **Suppress Map** are mutually exclusive; you cannot specify both.



If you want to use **Summary Only**, but you also want to advertise an individual route, then you create a [BGP Filtering Profile](#) that includes an [Unsuppress Map](#) route map that matches on the individual route.

5. Select **AS Set** to advertise the prefix with the list of AS numbers that make up the aggregate route; default is disabled.
6. Select **Aggregate Same MED Only** to cause route aggregation only if routes have the same Multi-Exit Discriminator (MED) values; default is enabled.
7. Select the **Type** of aggregate route: **IPv4** or **IPv6**.
8. Calculate the routes you want to summarize and then enter the **Summary Prefix** that spans those routes, by specifying an IP address/netmask or address object.
9. To prevent individual routes from being aggregated (suppress the aggregation), select a **Suppress Map** route map or [create a new BGP route map](#) whose match criterion specifies an IPv4 or IPv6 address Access List or Prefix List that includes those routes; default is **None**.



Remember that the purpose of the **Suppress route map** is to prevent certain routes from being aggregated in an advertisement. Therefore, in the route map you **permit** the routes that you want to suppress from being aggregated (you don't **deny** the routes that you want to suppress from being aggregated).



Summary Only and **Suppress Map** are mutually exclusive; you cannot specify both.

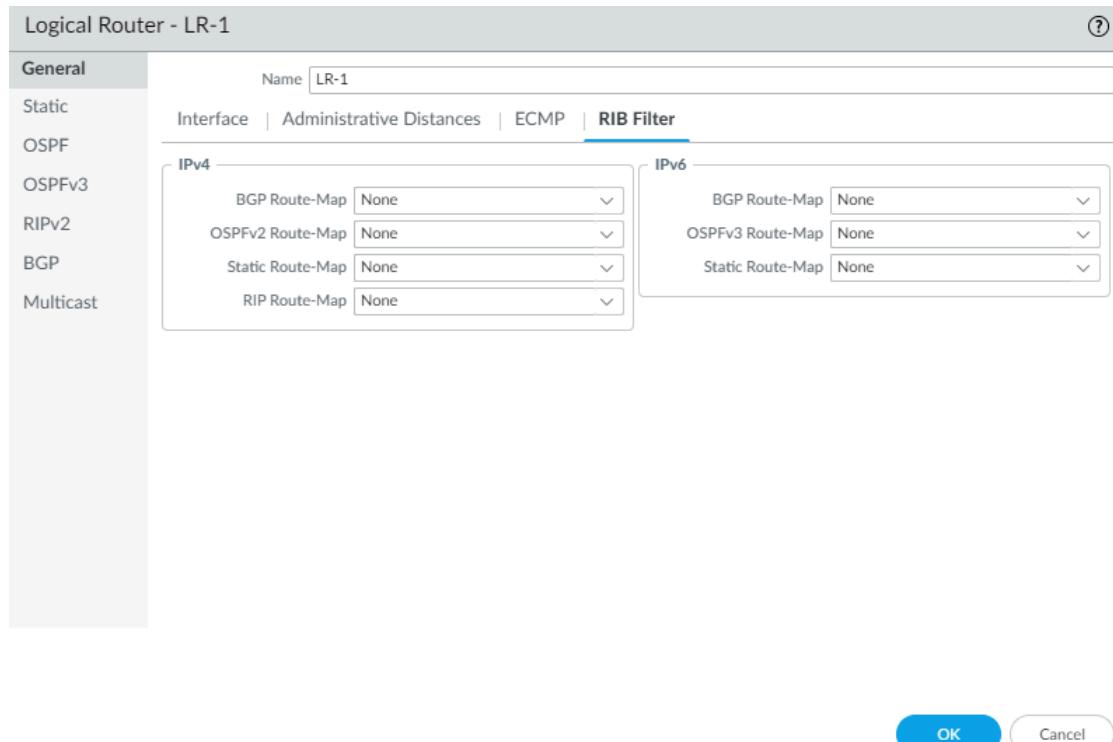
10. To set attribute information for the **Summary Prefix** (which has no attributes because you just created that combination of routes), select an **Attribute Map** route map or [create a new BGP route map](#) and set the attributes for the **Summary Prefix** (no match criteria). If there is no route map (**None**), the **Summary Prefix** will have the default attributes. Default is **None**.

STEP 8 | Click **OK**.

STEP 9 | (Optional) Control BGP routes that are placed in the global RIB.

You might learn routes and redistribute them, but not want them in the protocol's local route table or global RIB. You might want to add only specific routes to the global RIB.

1. Select **Network > Routing > Logical Routers** and select a logical router.
2. Select **RIB Filter** to allow routes into or prevent routes from being added to the global RIB.



3. To filter IPv4 BGP routes, in the IPv4 area, for **BGP Route-Map**, select a Redistribution Route Map or [create a new one](#).
4. To filter IPv6 BGP routes, in the IPv6 area, for **BGP Route-Map**, select a Redistribution Route Map or [create a new one](#).
5. Click **OK**.

Create BGP Routing Profiles

On an Advanced Routing Engine, BGP has many settings that you can easily configure in a profile and then apply to a BGP peer group or peer or to redistribution rules. Reuse profiles to apply them to multiple logical routers and virtual systems. Create multiple profiles of the same type to handle different peer groups and peers differently. BGP peer groups and peers inherit global profiles; you can also create a profile for a BGP peer group to override the global profile, and create a profile for a BGP peer, which overrides the profile for the peer group to which the peer belongs.

This topic describes the BGP routing profiles and how to create them.

- **BGP Authentication Profiles**—Specify the Secret key for MD5 authentication, which is used between BGP peers during negotiation to determine whether they can communicate with each other. Reference the profile in a BGP peer group or peer configuration.
- **BGP Timer Profiles**— Control various BGP timers that affect keepalive and update messages that advertise routes. Reference the profile in a BGP peer group or peer configuration.
- **BGP Address Family Profiles**—Determine the behavior of IPv6 or IPv4 when a BGP autonomous system uses both types of address. Reference the profile in a BGP peer group or peer configuration.
- **BGP Dampening Profiles**—Determine how to penalize a flapping route to suppress it from being used until it stabilizes. Reference the profile in a BGP peer group or peer configuration.
- **BGP Redistribution Profiles**—Redistribute static, connected, OSPF, OSPFv3, or RIP routes (that meet the criteria of the assigned route map) into BGP and apply the route map Set attributes to the redistributed routes. Reference the profile on **Network > Routing > Logical Routers > BGP > Redistribution**.
- **BGP Filtering Profiles**—Simultaneously apply multiple filters to a peer group or peer to do the following:
 - Accept routes that came from a specific AS Path (based on the AS Path access list).
 - Advertise routes that have a specific AS Path (based on the AS Path access list).
 - Accept routes to the local BGP RIB based on either a distribute list or prefix list (not both in the same Filtering Profile). A distribute list is based on source IP address with wildcard mask to get a prefix range. A prefix list is based on network address/prefix length.
 - Advertise routes from the local BGP RIB based on a distribute list or prefix list (not both in the same Filtering Profile).
 - Accept routes that meet route map attribute criteria into the local BGP RIB, and optionally set attributes.
 - Advertise routes that meet route map attribute criteria, and optionally set attributes.
 - Conditionally advertise routes that exist (satisfy exist criteria).
 - Conditionally advertise routes other than those that meet criteria (satisfy non-exist criteria).
 - Unsuppress dampened or summarized routes.

STEP 1 | Create a BGP Authentication profile.

1. Select **Network > Routing > Routing Profiles > BGP**.

The screenshot shows the PA-VM interface with the 'NETWORK' tab selected. On the left, there's a navigation tree with 'BGP' selected under 'Routing Profiles'. The main pane has two sections: 'BGP Auth Profiles' and 'BGP Timer Profiles'. The 'BGP Auth Profiles' section has a search bar and a table with one row ('default'). The 'BGP Timer Profiles' section also has a search bar and a table with one row ('default') showing values for Keep Alive (30), Hold (90), and MRAI (30). Buttons for '+ Add' and 'Delete' are visible in both sections.

2. **Add a BGP Auth Profile by Name** (a maximum of 63 characters) to identify the profile. The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Enter the **Secret** and **Confirm Secret**. The Secret is used as a key in MD5 authentication. The Secret can include only !@#%^_- characters and alphanumeric characters.
4. Click **OK**.

STEP 2 | Create a BGP Timer profile.

1. Select **Network > Routing > Routing Profiles > BGP**.
2. In the BGP Timer Profiles window, select the **default** BGP Timer Profile to see the default profile settings:

This is a configuration dialog titled 'BGP Timer Profile'. It contains fields for various timer parameters:

- Name: default
- Keep Alive Interval (sec): 30
- Hold Time (sec): 90
- Reconnect Retry Interval: 15
- Open Delay Time (sec): 0
- Minimum Route Advertise Interval (sec): 30

At the bottom are 'OK' and 'Cancel' buttons.

3. If the default BGP Timer Profile settings are not what you need, **Add a BGP Timer Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
4. Set the **Keep Alive Interval (sec)**—the interval, in seconds, at which the BGP speaker sends Keepalives to the peer (range is 0 to 1,200; default is 30). If no Keepalive is received from a peer during a Hold Time interval, the BGP peering is closed. Often, the

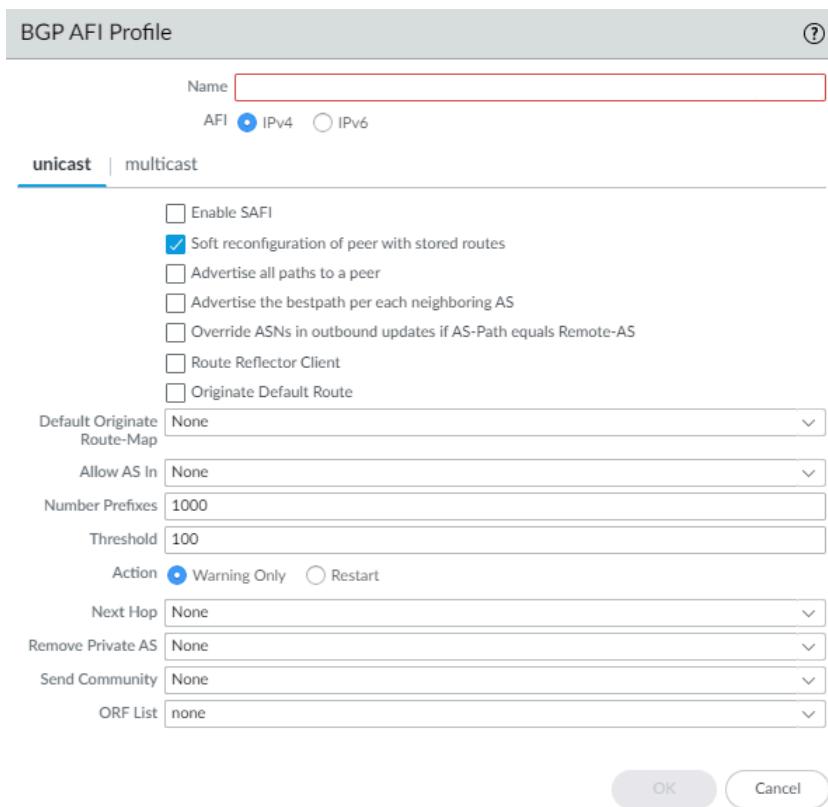
Hold Time is three times the Keep Alive Interval to allow for three missed Keepalives before BGP peering is brought down.

5. Set the **Hold Time (sec)**—the length of time, in seconds, that may elapse between successive Keepalive or Update messages from the peer before the peer connection is closed (range is 3 to 3,600; default is 90).
6. Set the **Reconnect Retry Interval**—the number of seconds to wait in Idle state before retrying to connect to the peer (range is 1 to 3,600; default is 15).
7. Set the **Open Delay Time (sec)**—the number of seconds of delay between opening the TCP connection to the peer and sending the first BGP Open message to establish a BGP connection (range is 0 to 240; default is 0).
8. Set the **Minimum Route Advertise Interval (sec)**—the minimum amount of time, in seconds, that must elapse between an advertisement and/or withdrawal of routes to a particular destination by a BGP speaker to a peer (range is 1 to 600; default is 30).
9. Click **OK**.

STEP 3 | To use **MP-BGP**, create a BGP Address Family Identifier (AFI) profile of shared attributes.

1. Select **Network > Routing > Routing Profiles > BGP**.
2. **Add a BGP Address Family Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain

a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.



3. Select **IPv4** or **IPv6** AFI to specify the type of profile.
4. Select **unicast** or **multicast**.



Multicast is supported only for an **IPv4** AFI profile.

5. On the **unicast** tab, **Enable SAFI** to enable the unicast SAFI for the profile. On the **multicast** tab, **Enable SAFI** to enable the multicast SAFI for the profile. If **Enable SAFI** is checked for both **unicast** and **multicast**, both SAFI are enabled. At least one SAFI must be enabled for the BGP profile to be valid.
6. Select **Soft reconfiguration of peer with stored routes** to cause the firewall to perform a soft reset of itself after settings of any of its BGP peers are updated. (Enabled by default.)
7. **Advertise all paths to peers**— to have BGP advertise all known paths to neighbor in order to preserve multipath capabilities inside a network.
8. **Advertise the bestpath for each neighboring AS** to have BGP advertise the best known paths to neighbors in order to preserve multipath capabilities inside a network. Disable this if you want to advertise the same path to all autonomous systems.
9. **Override ASNs in outbound updates if AS-Path equals Remote-AS**—This setting is helpful if you have multiple sites belonging to the same AS number (AS 64512, for example) and there is another AS between them. A router between the two sites receives an Update advertising a route that can access AS 64512. To avoid the second

site dropping the Update because it is also in AS 64512, the intermediate router replaces AS 64512 with its own AS number (ASN), AS 64522, for example.

10. **Enable Route Reflector Client** to make the BGP peer a Route Reflector client in an IBGP network.
11. **Originate Default Route**—Select to generate a default route and place it in the local BGP RIB.
12. **Default Originate Route-Map**—Select or create a route map to control the attributes of the default route.
13. **Allow AS in:**
 - **Origin**—Accept routes even if the firewall's own AS is present in the AS_PATH.
 - **Occurrence**—Number of times the firewall's own AS can be in the AS_PATH.
 - **None**—(default setting) No action taken.
14. **Number Prefixes**—Maximum number of prefixes to accept (learn) from the peer. Range is 1 to 4,294,967,295; default is 1,000.
15. **Threshold**—Percentage of the maximum number of prefixes. The prefixes are added to the BGP local RIB. If the peer advertises more than the threshold, the firewall takes the specified action (**Warning Only** or **Restart**). Range is 1 to 100; default is 100.
16. **Action**—**Warning Only** message in system logs or **Restart** the BGP peer connection after the maximum number of prefixes is exceeded.
17. Select the **Next Hop**:
 - **Self**—Causes the firewall to change the Next Hop address (in Updates it receives) to its own IP address in the Update before sending it on. This is helpful when the firewall is communicating with an EBGP router (in another AS) and with an IBGP router (in its own AS). For example, suppose the Next Hop address in a BGP Update that arrives at AS 64512 is the IP address of the egress interface of Router 2 where the Update egressed AS 64518. The Update indicates that to reach networks that Router 2 is advertising, use the Next Hop address of Router 2. However, if the firewall sends that Update to an iBGP neighbor in AS 64512, the unchanged Next Hop of Router 2 is outside AS 64512 and the iBGP neighbor does not have a route to it. When you select **Self**, the firewall changes the Next Hop to its own IP address so that an iBGP

neighbor can use that Next Hop to reach the firewall, which in turn can reach the eBGP router.

- **Self Force**—Force set the next hop to self for the reflected routes.
 - **None**—(default setting) Keep the original Next Hop in the attribute.
18. To have BGP remove private AS numbers from the AS_PATH attribute in Updates that the firewall sends to a peer in another AS, in **Remove Private AS**, select one of the following:
- **All**—Remove all private AS numbers.
 - **Replace AS**—Replace all private AS numbers with the firewall's AS number.
 - **None**—(default setting) No action taken.
19. For **Send Community**, select the type of BGP community attribute to send in outbound Update packets:
- **All**—Send all communities.
 - **Both**—Send standard and extended communities.
 - **Extended**—Send extended communities ([RFC 4360](#)).
 - **Large**—Send large communities ([RFC 8092](#)).
 - **Standard**—Send standard communities ([RFC 1997](#)).
 - **None**—(default setting) Do not send any communities.
20. For **ORF List**—Advertise the ability of the peer group or peer to send a prefix list and/or receive a prefix list to implement outbound route filtering (ORF) at the source, and thereby minimize sending or receiving unwanted prefixes in Updates. Select an ORF capability setting:
- **none**—(default setting) The peer group or peer (where this AFI profile is applied) has no ORF capability.
 - **both**—Advertise that the peer group or peer (where this AFI profile is applied) can **send** a prefix list and **receive** a prefix list to implement ORF.
 - **receive**—Advertise that the peer group or peer (where this AFI profile is applied) can receive a prefix list to implement ORF. The local peer receives the remote peer's ORF capability and prefix list, which it implements as an outbound route filter.
 - **send**—Advertise that the peer group or peer (where this AFI profile is applied) can send a prefix list to implement ORF. The remote peer (with receive capability)

receives the ORF capability and implements the prefix list it received as an outbound route filter when advertising routes to the sender.

ORF is a solution to two potential problems: a) wasting bandwidth by advertising unwanted routes and b) filtering route prefixes that perhaps the receiving peer wants. Implement ORF by doing the following:

1. Specify ORF capability in the Address Family profile.
2. For a peer group or peer that is a sender (**send** or **both** capability), create a prefix list containing the set of prefixes the peer group/peer wants to receive.
3. Create a BGP Filtering profile and in the Inbound Prefix List, select the prefix list you created.
4. For the BGP peer group, select the Address Family profile you created to apply it to the peer group. In the case of the sender, also select the Filtering Profile you created (which indicates the prefix list). If the peer group or peer is an ORF receiver only, it does not need the Filtering Profile; it needs only the Address Family profile to indicate ORF **receive** capability.

21. Click **OK**.

STEP 4 | Create a BGP Dampening Profile.

1. Select **Network > Routing > Routing Profiles > BGP**.
2. **Add a BGP Dampening Profile by Name.** The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Enter a helpful **Description**.
4. **Suppress Limit**—Enter the suppress value (cumulative value of the penalties for flapping), at which point all the routes coming from a peer are dampened. Range is 1 to 20,000; default is 2,000.
5. **Reuse Limit**—Enter the value that controls when a route can be reused based on the procedure described for **Half Life**. Range is 1 to 20,000; default is 750.
6. **Half Life (min)**—Enter the number of minutes for the half life time to control the stability metric (penalty) applied to a flapping route. Range is 1 to 45; default is 15. The stability metric starts at 1,000. After a penalized route stabilizes, the half life timer counts down until it expires, at which point the next stability metric applied to the route is only half of

the previous value (500). Successive cuts continue until the stability metric is less than half of the Reuse Limit, and then the stability metric is removed from the route.

7. **Maximum Suppress Time (min)**—Enter the maximum number of minutes a route can be suppressed, regardless of how unstable it has been. Range is 1 to 255; default is 60.

The dialog box is titled "BGP Dampening Profile". It contains the following fields:

- Name: [redacted]
- Description: [redacted]
- Suppress Limit: 2000
- Reuse Limit: 750
- Half Life (min): 15
- Maximum Suppress Time (min): 60

At the bottom are "OK" and "Cancel" buttons.

8. Click **OK**.

STEP 5 | Create a BGP Redistribution Profile to redistribute static, connected, and OSPF routes (that match the corresponding route map) to BGP.

1. Select **Network > Routing > Routing Profiles > BGP**.
2. **Add a BGP Redistribution Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Select the **AFI** of routes to redistribute: **IPv4** or **IPv6**.

The dialog box is titled "BGP Redistribution Profile". It contains the following fields:

- Name: [redacted]
- AFI: IPv4 IPv6
- Static**:
 - Enable:
 - Metric: [1 - 65535]
 - Route-Map: None
- OSPF**:
 - Enable:
 - Metric: [1 - 65535]
 - Route-Map: None
- Connected**:
 - Enable:
 - Metric: [1 - 65535]
 - Route-Map: None
- RIP**:
 - Enable:
 - Metric: [1 - 65535]
 - Route-Map: None

At the bottom are "OK" and "Cancel" buttons.

4. Select **Static** to configure static route redistribution.
5. **Enable** redistribution of IPv4 or IPv6 static routes (based on the AFI you selected).
6. Configure the **Metric** to apply to the static routes being redistributed into BGP (range is 1 to 65,535).
7. Select a **Route-Map** to specify the match criteria that determine which static routes to redistribute. Default is **None**. If the route map Set configuration includes a Metric

- Action and Metric Value, they are applied to the redistributed route. If no Metric Value is configured in the route map, the firewall applies default metric values.
8. Select **Connected** to configure connected route redistribution.
 9. **Enable** redistribution of locally connected IPv4 or IPv6 routes (based on the AFI you selected).
 10. Configure the **Metric** to apply to the connected routes being redistributed into BGP (range is 1 to 65,535).
 11. Select a **Route Map** to specify the match criteria that determine which connected routes to redistribute. Default is None. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route.
 12. (**IPv4 AFI only**) Select **OSPFv2** to configure OSPFv2 route redistribution.
 13. **Enable** redistribution of OSPFv2 routes.
 14. Configure the **Metric** to apply to the OSPF routes being redistributed into BGP (range is 1 to 65,535).
 15. Select a **Route-Map** to specify the match criteria that determine which OSPF routes to redistribute. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route.
 16. (**IPv4 AFI only**) Select **RIPv2** to configure RIPv2 route redistribution.
 17. **Enable** redistribution of RIPv2 routes.
 18. Configure the **Metric** to apply to the RIP routes being redistributed into BGP (range is 1 to 65,535).
 19. Select a **Route-Map** to specify the match criteria that determine which RIP routes to redistribute. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route.
 20. (**IPv6 AFI only**) Select **OSPFv3** to configure OSPFv3 route redistribution.
 21. **Enable** redistribution of OSPFv3 routes.
 22. Configure the **Metric** to apply to the OSPFv3 routes being redistributed into BGP (range is 1 to 65,535).
 23. Select a **Route-Map** to specify the match criteria that determine which OSPFv3 routes to redistribute. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route.
 24. Click **OK**.

STEP 6 | Create a BGP Filtering Profile.

1. Select **Network > Routing > Routing Profiles > BGP**.
2. **Add a BGP Filtering Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a

combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.

3. Enter a helpful **Description**.
4. Select **IPv4 or IPv6** Address Family Identifier (AFI) to indicate the type of route to filter.

The screenshot shows the 'BGP Filtering Profile' configuration dialog. At the top, there are fields for 'Name' and 'Description', both currently empty. Below these is a radio button group for 'AFI' with 'IPv4' selected and 'IPv6' unselected. The 'Multicast' tab is active, indicated by a blue underline. Under the 'Multicast' tab, there is a checkbox for 'Inherit from Unicast'. Below this are sections for 'Network Filter' (Inbound and Outbound), 'Conditional Advertisement' (Exist and Non-Exist), and an 'Unsuppress Map' section. At the bottom right are 'OK' and 'Cancel' buttons.

5. Select **Unicast** or **Multicast** Subsequent Address Family Identifier (SAFI).
6. For **Unicast**, **Inbound Filter List** – Select an AS Path access list or [create a new AS Path access list](#) to specify that, when receiving routes from peers, only routes with the same AS Path are imported from the peer group or peer, meaning added to the local BGP RIB.
7. In the Network Filter area, **Inbound–Distribute List** – Use an access list (Source Address only; not Destination Address) to filter BGP routing information that BGP receives. Mutually exclusive with Inbound Prefix List in a single Filtering Profile.
8. **Prefix List** – Use a prefix list to filter BGP routing information that BGP receives, based on a network prefix. Mutually exclusive with Inbound Distribute List in a single Filtering Profile.
9. **Inbound Route Map** – Use a route map to have even more control over which routes are allowed into the local BGP RIB (Match criteria) and to set attributes for the routes (Set

options). For example, you can control route preference by prepending an AS to the AS Path of a route.



If an Inbound Route Map is configured with an Inbound Distribute List or Prefix List, the conditions of both the route map and list must be met (logical AND).

10. **Outbound Filter List**—Select an AS Path access list or [create a new AS Path access list](#) to specify that only routes with the same AS Path are advertised to a peer router (peer group or peer where this filter is applied).
11. **Outbound—Distribute List**—Use an access list to filter BGP routing information that BGP advertises, based on the IP address of the destination. Mutually exclusive with Outbound Prefix List in a single Filtering Profile.
12. **Prefix List**—Use a prefix list to filter BGP routing information that BGP advertises, based on a network prefix. Mutually exclusive with Outbound Distribute List in a single Filtering Profile.
13. **Outbound Route Map**—Use a route map to have even more control over which routes BGP advertises (Match criteria) and to set attributes for advertised routes.

*If an Outbound Route Map is configured with an Outbound Distribute List or Prefix List, the conditions of both the route map and list must be met (logical AND).*
14. Configure conditional advertisements, which allow you to control what route to advertise in the event that a different route exists or does not exist in the local BGP RIB. A route not existing in the local BGP RIB can indicate a peering or reachability failure. Conditional advertisements are useful in cases where you want to try to force routes to one AS over another, such as when you have links to the internet through multiple ISPs and you want traffic to be routed to one provider instead of another, except when there is a loss of connectivity to the preferred provider. In the Conditional Advertisement **Exist** area:
 - Select or create a route map in the **Exist Map** field to specify the match criteria for the conditional advertisement. Only the Match portion of the route map in this field is considered; the Set portion is ignored.
 - **Advertise Map**—Select or create a route map to specify the routes to advertise in the event that the condition is met (routes from the Exist Map exist in the local BGP RIB). Only the Match portion of the route map in this field is considered; the Set portion is ignored.
15. In the Conditional Advertisement **Non-Exist** area:
 - Select or create a route map in the **Non-Exist Map** field to specify the match criteria for which routes do not exist in the local BGP RIB in order to conditionally advertise. Only the Match portion of the route map in this field is considered; the Set portion is ignored.
 - **Advertise Map**—Select or create a route map to specify routes to advertise when the routes in the Non-Exist Map are not in the local BGP RIB. Only the Match portion of the route map in this field is considered; the Set portion is ignored.
16. **Unsuppress Map**—Select or create a route map of routes that you want to unsuppress, perhaps because they were summarized and therefore suppressed, or they were

suppressed because they met dampening criteria, but you want specific routes to be advertised (unsuppressed).

17. **(IPv4 AFI only)** Select **Multicast** to filter MP-BGP Multicast routes. Select **Inherit from Unicast** if you want all filtering from the Unicast SAIFI to also apply to the Multicast SAIFI. Otherwise, continue to configure the following filtering fields.
18. For **Multicast, Inbound Filter List**—Specify an AS Path access list or [create a new AS Path access list](#) to specify that, when receiving routes from peers, only routes with the same AS Path are imported from the peer group or peer, meaning added to the local BGP RIB.
19. In the Network Filter area, **Inbound—Distribute List**—Use an access list (Source Address only; not Destination Address) to filter BGP routing information that BGP receives. Mutually exclusive with Inbound Prefix List in a single Filtering Profile.
20. **Prefix List**—Use a prefix list to filter BGP routing information that BGP receives, based on a network prefix. Mutually exclusive with Inbound Distribute List in a single Filtering Profile.
21. **Inbound Route Map**—Use a route map to have even more control over which routes are allowed into the local BGP RIB (Match criteria) and to set attributes for the routes (Set options). For example, you can control route preference by prepending an AS to the AS Path of a route.



If an Inbound Route Map is configured with an Inbound Distribute List or Prefix List, the conditions of both the route map and list must be met (logical AND).

22. **Outbound Filter List**—Select an AS Path access list or [create a new AS Path access list](#) to specify that only routes with the same AS Path are advertised to a peer router (peer group or peer where this filter is applied).
23. **Outbound—Distribute List**—Use an access list to filter BGP routing information that BGP advertises, based on the IP address of the destination. Mutually exclusive with Outbound Prefix List in a single Filtering Profile.
24. **Prefix List**—Use a prefix list to filter BGP routing information that BGP advertises, based on a network prefix. Mutually exclusive with Outbound Distribute List in a single Filtering Profile.
25. **Outbound Route Map**—Use a route map to have even more control over which routes BGP advertises (Match criteria) and to set attributes for advertised routes.



If an Outbound Route Map is configured with an Outbound Distribute List or Prefix List, the conditions of both the route map and list must be met (logical AND).

26. Configure conditional advertisements, which allow you to control what route to advertise in the event that a different route exists or does not exist in the local BGP RIB. A route not existing in the local BGP RIB can indicate a peering or reachability failure. Conditional advertisements are useful in cases where you want to try to force routes to one AS over another, such as when you have links to the internet through multiple ISPs and you want traffic to be routed to one provider instead of another, except when there

is a loss of connectivity to the preferred provider. In the Conditional Advertisement **Exist** area:

- Select or create a route map in the **Exist Map** field to specify the match criteria for the conditional advertisement. Only the Match portion of the route map in this field is considered; the Set portion is ignored.
- **Advertise Map**—Select or create a route map to specify the routes to advertise in the event that the condition is met (routes from the Exist Map exist in the local BGP RIB). Only the Match portion of the route map in this field is considered; the Set portion is ignored.

27. In the Conditional Advertisement **Non-Exist** area:

- Select or create a route map in the **Non-Exist Map** field to specify the match criteria for which routes do not exist in the local BGP RIB in order to conditionally advertise. Only the Match portion of the route map in this field is considered; the Set portion is ignored.
- **Advertise Map**—Select or create a route map to specify routes to advertise when the routes in the Non-Exist Map are not in the local BGP RIB. Only the Match portion of the route map in this field is considered; the Set portion is ignored.

28. **Unsuppress Map**—Select or create a route map of routes that you want to unsuppress, perhaps because they were summarized and therefore suppressed, or they were suppressed because they met dampening criteria, but you want specific routes to be advertised (unsuppressed).

29. Click **OK**.

Create Filters for the Advanced Routing Engine

The Advanced Routing Engine supports the filters described in this topic. Access lists, prefix lists, and redistribution route maps can apply to BGP, OSPFv2, OSPFv3 and RIPv2. Access lists and prefix lists can also apply to IPv4 multicast. Multicast route maps apply to IPv4 multicast. AS path access lists, community lists, and BGP route maps apply to BGP only.

Create a filter and reference the filter in a profile or other appropriate location to easily and consistently apply settings that control such things as route acceptance from peers into the local RIB, route advertisements to peers, conditional advertisements, setting attributes, exporting and importing routes to and from other routers, route aggregation, and route redistribution.

- **Access Lists**—Use an access list:

- To filter network routes based on IPv4/IPv6 source addresses and IPv4 destination addresses. For IPv4 access lists, source and destination addresses can be specified by an address and wildcard mask to express a range of addresses. IPv6 access lists can specify source addresses and subnet.
- In a BGP Filtering profile, specify an Inbound Distribute List (access list) to control which routes BGP will accept from a peer group or peer (neighbor). This means that routes matching a deny access list rule are not placed in the local BGP RIB; routes matching a permit access list rule are placed in the local BGP RIB. You apply the BGP Filtering profile to a BGP peer group or peer in the Filtering IPv4 Unicast or Filtering IPv6 Unicast field. (To do this for a peer, select **Inherit No**). Peer settings take precedence over peer group settings.
- In a BGP Filtering profile, specify an Outbound Distribute List (access list) to control which routes the firewall advertises to its peer group or peer, based on your network and BGP deployment. Then apply the BGP Filtering profile to a BGP peer group or peer in the Filtering IPv4 Unicast or Filtering IPv6 Unicast field. (To do this for a peer, select **Inherit No**). Peer settings take precedence over peer group settings.
- As match criteria in a Redistribution route map to specify IPv4 or IPv6 destination Addresses, Next Hop, or Route Source.
- In a BGP route map as match criteria for an IPv4 Address, Next Hop, or Route Source, and also for an IPv6 Address.
- In OSPFv2 and OSPFv3 Import Lists and Export Lists for an Area Border Router (ABR).
- To specify PIM group permissions for IPv4 multicast.



An access list is not for filtering user traffic or for providing security.

An access list can have multiple rules; routes are evaluated against the rules in sequential order. When a route matches a rule, the deny or permit action occurs and the route is not evaluated against subsequent rules.

The aggregated view displays all configured access lists; you can highlight an access list to then modify or delete it.

- **Prefix Lists**—Use a prefix list:
 - To filter network routes that are added to a local RIB based on route prefix and prefix length.
 - In a BGP Filtering Profile, specify an Inbound Prefix List to control which routes BGP will accept from a peer group or peer (neighbor). This means that routes matching a deny prefix list rule are not placed in the local BGP RIB; routes matching a permit prefix list rule are placed in the local BGP RIB. Then apply the BGP Filtering profile to a BGP peer group in the Filtering IPv4 Unicast or Filtering IPv6 Unicast field. (To do this for a peer, select Inherit No). Peer settings take precedence over peer group settings.
 - In a BGP Filtering profile, specify an Outbound Prefix List to control which routes the firewall advertises to its peer group or peer, based on your network and BGP deployment. Then apply the BGP Filtering profile to a BGP peer group or peer in the Filtering IPv4 Unicast or Filtering IPv6 Unicast field. (To do this for a peer, select Inherit No). Peer settings take precedence over peer group settings.
 - As match criteria in a Redistribution route map to specify IPv4 or IPv6 destination Addresses, Next Hop, or Route Source.
 - In a BGP route map as match criteria for an IPv4 Address, Next Hop, or Route Source, and also for an IPv6 Address.
 - For an OSPFv2 or OSPFv3 ABR of an area, in an Inbound Filter List or Outbound Filter List.
 - In an IPv4 Multicast PIM general configuration to specify an SPT threshold.
 - In an IPv4 Multicast route map.

A prefix list can have multiple rules; routes are evaluated against the rules in sequential order. When a route matches a rule, the deny or permit action occurs and the route is not evaluated against subsequent rules. A prefix list is flexible in that it allows you to configure a prefix with a prefix length (that together identify the prefix), and also have a range by specifying that the prefix length be greater than, less than, or equal to a value. The firewall evaluates prefix lists more efficiently than access lists.

- **Redistribution Route Maps**—Use a Redistribution Route Map in a Redistribution Profile to specify which BGP, OSPFv2, OSPFv3, RIP, connected or static routes (the source protocol) to redistribute to BGP, OSPFv2, OSPFv3, RIP, or the local RIB (the destination protocol). You can also redistribute BGP host routes to BGP peers. The match criteria can include IPv4 and IPv6 addresses specified by an access list and prefix list.

A Redistribution route map can have multiple entries; routes are evaluated against the entries in sequential order. When a route matches an entry, it is permitted or denied and the route is not evaluated against subsequent entries. If the action of the matching entry is Permit, the firewall also sets the configured attributes from the route map to the redistributed route.

- **Multicast Route Maps**—Create a multicast route map to filter sources for a dynamic IGMP interface.

The following filters apply to BGP only.

- **AS Path Access Lists**—Create an AS Path access list:
 - To control importing of BGP routes (into the local BGP RIB) that came from another router, use in a BGP Filtering Profile, in the Inbound Filter List. For example, you want to import only routes that came through specific autonomous systems.
 - To control exporting of BGP routes to another router, use in a BGP Filtering Profile, in the Outbound Filter List.
 - To do anything a BGP route map can do, use in a BGP route map as a match criterion.
 - To redistribute BGP routes, use in a BGP Redistribution route map (AS Path) as a match criterion.

An AS Path access list can have a maximum of 64 rules and ends with an implicit **Permit Any** rule. Use an AS Path access list to deny autonomous systems. Routes are evaluated against the rules in sequential order. When a route matches a rule, the deny or permit action occurs and the route is not evaluated against subsequent rules.

- **Community Lists**—Create a community list:
 - To reference in a BGP route map to match on BGP community attributes of routes that you want to control in some way. For example, you can set a group of routes (that share a community attribute) to have a specific metric or local preference.
 - To reference in the set actions of a BGP route map to remove communities from routes that meet the match criteria.
 - To match BGP communities in routes that you want to redistribute using a Redistribution route map.
- **BGP Route Maps**—Create a BGP route map:
 - For the **Default Originate Route-Map** field of a BGP AFI Profile; the match criteria define when to generate the default route (0.0.0.0). Apply the BGP AFI profile to a BGP peer group or peer. The Match criteria can be any parameter and if there is a match to an existing BGP route, the default route is created; the Set portion of the route map is not used. Instead, you can use an outbound route-map to set properties for the generated default route.
 - To set (override) BGP attributes that BGP is sending to a peer.
 - For NAT, to set Source Address and IPv4 Next Hop for a certain group of prefixes you are advertising, enter a public IP address from the NAT pool to replace a private IP address.
 - To redistribute static, connected, or OSPF routes into BGP; then reference the BGP route map in a BGP Redistribution profile.
 - In a BGP Filtering Profile, use a BGP route map in **Inbound Route Map** or **Outbound Route Map** to filter routes that are accepted (learned) from BGP peers into the local BGP RIB (inbound) or advertised to BGP peers (outbound).
 - To conditionally advertise BGP routes, in a BGP Filtering Profile, create an **Exist Map**, which specifies that if these conditions in the route exist, advertise the route based on an

Advertise Map. Alternatively, specify that if these conditions do not exist, advertise the route based on a **Non-Exist Advertise Map**.

- In a BGP Filtering Profile, set an IPv4 Next Hop to use a public NAT address rather than a private address.
- In a BGP Filtering Profile, use a BGP route map to unsuppress routes that were suppressed due to route dampening or aggregation.
- To conditionally filter more specific routes, for a logical router, configure BGP **Aggregate Routes** and provide the **Suppress Map**.
- To set attributes for an aggregate route, for a logical router, configure BGP **Aggregate Routes** and provide the **Attribute Map**.

A filter can have multiple rules; the firewall evaluates packets or routes against the rules in a filter in order by sequence number (**Seq**) of the rule. When a packet or route matches a rule, the deny or permit action occurs and the packet or route is not evaluated against subsequent rules.



*All filters except AS Path access lists end with an implicit **Deny Any** rule. All filters except for AS Path access lists must have at least one **Permit** rule; otherwise, all examined routes/packets are denied. AS Path access lists end with an implicit **Permit Any** rule.*

Select a configured **Seq** number to open a rule and modify it. Select an **Action** field in a configured rule to modify only the Permit or Deny action.



When adding a rule, leave enough unused sequence numbers between rules to allow future rules to be inserted in the filter. For example, use Seq numbers 10, 20, 30, etc.

STEP 1 | Create an access list to permit or deny IPv4 or IPv6 addresses where this filter is applied.

1. Select **Network > Routing > Routing Profiles > Filters**.
2. Add a **Filters Access List by Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Enter a helpful **Description**.
4. Select the **Type** of access list: **IPv4** or **IPv6**.
 1. For IPv4, Add an **IPv4 Entry** and enter the **Seq** number for the rule (range is 1 to 65,535).
 2. Select the **Action**: **Deny** (the default) or **Permit**.
 3. For **Source Address**, there are three options: select **Address** and in the subsequent **Address** field, enter an IPv4 address. Enter a **Wildcard** mask to indicate a range. A zero (0) in the mask indicates that bit must match the corresponding bit in the address; a one (1) in the mask indicates a “don’t care” bit. The other options are **Any** or **None**.
 4. For **Destination Address**, select **Address** and in the subsequent **Address** field, enter an IPv4 address. Enter a **Wildcard**. A zero (0) in the mask indicates a bit that must

match; a one (1) in the mask indicates a “don’t care” bit. The other options are **Any** or **None**.

5. Click **OK** to save the entry.

The screenshot shows a configuration window titled "Filters Access List". At the top, there are fields for "Name" (set to "filter_networks_to_allow") and "Description" (set to "permit 192.168.0.0 subnets"). Below these, a "Type" field has "IPv4" selected. The main area is a table titled "Entry" with columns: SEQ, ACTION, SRC NETWORK, WILDCARD, DST NETWORK, and WILDCARD. A single row is present with values: SEQ 5, ACTION permit, SRC NETWORK 192.168.2.1, WILDCARD 0.0.255.255, DST NETWORK none, and WILDCARD. At the bottom of the table are buttons for "+ Add" and "- Delete". Below the table are "OK" and "Cancel" buttons.

Entry	SEQ	ACTION	SRC NETWORK	WILDCARD	DST NETWORK	WILDCARD
	5	permit	192.168.2.1	0.0.255.255	none	

+ Add - Delete

OK Cancel

5. Alternatively, select the **Type** to be **IPv6**.

1. For IPv6, **Add an IPv6 Entry** and enter the **Seq** number (range is 1 to 65,535).
2. Select the **Action**: **Deny** (the default) or **Permit**.
3. For **Source Address**, there are three options: select **Address** and in the subsequent **Address** field, enter an **IPv6 Address**. Optionally select **Exact Match of this address** to have the firewall perform a comparison of both the prefix and prefix length and they must match exactly; otherwise, the firewall determines the match comparison based on whether the route is in the same subnet as the configured prefix. (If the Source

Address is **Any** or **None**, you cannot select **Exact Match of this address**.) The other options are **Any** or **None**.

4. Click **OK** to save the entry. Optionally add more entries.

Filters Access List

Entry	SEQ	ACTION	SRC NETWORK/MASK	EXACT MATCH

Name Description
Type IPv4 IPv6

+ Add **- Delete**

OK **Cancel**

6. Click **OK** to save the access list.

STEP 2 | Create a prefix list.

1. Select **Network > Routing > Routing Profiles > Filters**.
2. Add a **Filters Prefix List by Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Enter a helpful **Description**.
4. Select the **Type** of prefix for this rule to filter: **IPv4** or **IPv6**.

The screenshot shows the 'Filters Prefix List' configuration dialog. At the top, there are fields for 'Name' and 'Description'. Below that, a 'Type' section has 'IPv4' selected. The main area is a table titled 'Entry' with columns: SEQ, ACTION, NETWORK..., >= MAX PREFIX LENGTH, and <= MAX PREFIX LENGTH. At the bottom of the table are 'Add' and 'Delete' buttons. At the very bottom are 'OK' and 'Cancel' buttons.

1. For IPv4, Add an **IPv4 Entry**, and enter the **Seq** number for the rule; range is 1 to 65,535.
2. Select the **Action**: Deny (the default) or Permit.
3. For **Prefix**, there are three options; default is **None**. Another option is to select **Network any**. The third option is to select **Entry** and enter an IPv4 **Network** prefix with slash and a base prefix length that together specify a network, for example, 192.168.2.0/24. Optionally specify that the prefix length be **Greater Than Or Equal** to a number (that is at least as large as the base length you specified; range is 0 to 32). Optionally specify a top limit to the range by specifying **Less Than Or Equal** to a

number (that is at least as high as the base length and at least as high as the **Greater Than Or Equal** length if configured; range is 0 to 32).

The screenshot shows a configuration dialog titled "New IPv4 Entry". The "Seq" field is set to "1 - 65535". The "Action" field has "Deny" selected. The "Prefix" field contains "Entry". The "Network" field is a dropdown menu. Below it are two numerical input fields: "Greater Than Or Equal" set to "0 - 32" and "Less Than Or Equal" set to "0 - 32". At the bottom are "OK" and "Cancel" buttons.

Comparing a route to the prefix rule (IPv4 or IPv6) is a two-step process: 1) Match the prefix with the network first. 2) Match the prefix length to the mask range (Greater Than or Equal to Less Than Or Equal). For example, consider the prefix list rule with Network 192.168.3.0/24, and a prefix length Greater Than or Equal to 26 and Less Than or Equal to 30. The following table shows routes that are tested and whether they pass or fail the rule. Routes that pass the rule are subject to the configured action (Deny or Permit).

Sample Route	Result
192.168.3.0/28	Pass: the network and prefix length match the rule.
192.168.2.0/30	Fail: network does not match the rule.
192.168.3.0/32	Fail: prefix length does not match the rule.

In the output summary of the rule, LOU is Logical Operator Unit (equal, greater or equal, less or equal). \geq indicates a prefix length greater than or equal to the value; it is the lowest value of a range of the prefix length. \leq indicates a prefix length less than or equal to the value; it is the highest value of a range of the prefix length.

5. Alternatively, **Add an IPv6 Entry** and follow the steps similar to those for an IPv4 prefix rule. The range of the IPv6 prefix length is **Greater Than or Equal** to 0 to 128 and **Less Than Or Equal** to 0 to 128.

For example, consider the prefix list rule with Network 2001:db8:1/48, and a prefix length Greater Than or Equal to 56 and Less Than or Equal to 64. The following table shows routes that are tested and whether they pass or fail the rule. Routes that pass the rule are subject to the configured action (Deny or Permit).

Sample Route	Result
2001:db8:1/64	Pass: the network and prefix length match the rule.
2001:db8:2/48	Fail: network does not match the rule.

Sample Route	Result
2001:db8:1/65	Fail: prefix length does not match the rule.

6. Click **OK** to save the prefix entry. Optionally add more entries.
7. Click **OK** to save the Prefix List.

STEP 3 | Create an AS Path Access List for BGP.

1. Select **Network > Routing > Routing Profiles > Filters**.
2. **Add an AS Path Acess List by Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Enter a helpful **Description**.
4. **Add an Entry** and enter a **Seq** number; range is 1 to 65,535.
5. Select the **Action: Deny** (the default) or **Permit**.



*Each AS Path access list ends with an implicit **Permit Any** rule. Use an AS Path access list to deny autonomous systems.*

6. Enter the **Aspath Regex** (regular expression) in the format **regex1:regex2:regex3**, where a colon (:) separates three AS values. Characters allowed are 1234567890_^.{}[]\$*+.?-\. For example, .*65000 in a Deny statement excludes prefixes originating from AS 65000.

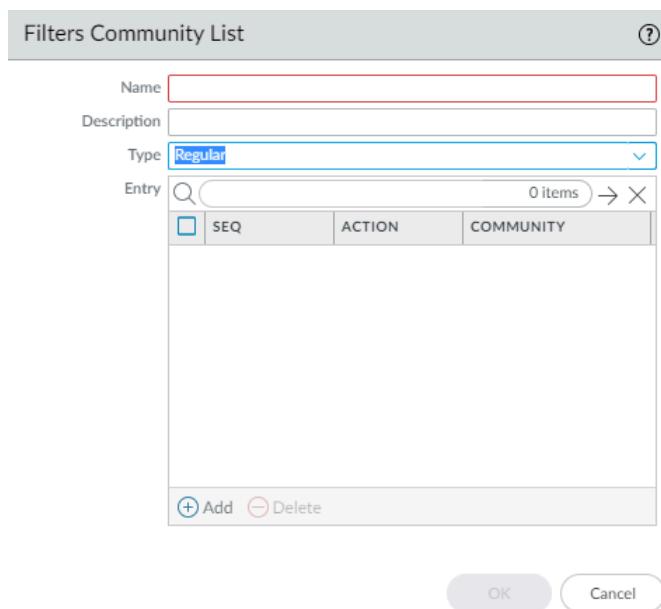
Filters AS Path Access List (?)

Name	<input type="text"/>		
Description	<input type="text"/>		
Entry	SEQ	ACTION	REGULAR EXPRESSION
+ Add - Delete			
OK Cancel			

7. Click **OK** to save the entry. Optionally add more entries; a maximum of 64 entries are allowed in an AS Path access list.
8. Click **OK** to save the AS Path access list.

STEP 4 | Create a Community List.

1. Select **Network > Routing > Routing Profiles > Filters**.
2. Add a **Filters Community List by Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Enter a helpful **Description**.



4. Select the **Type**:

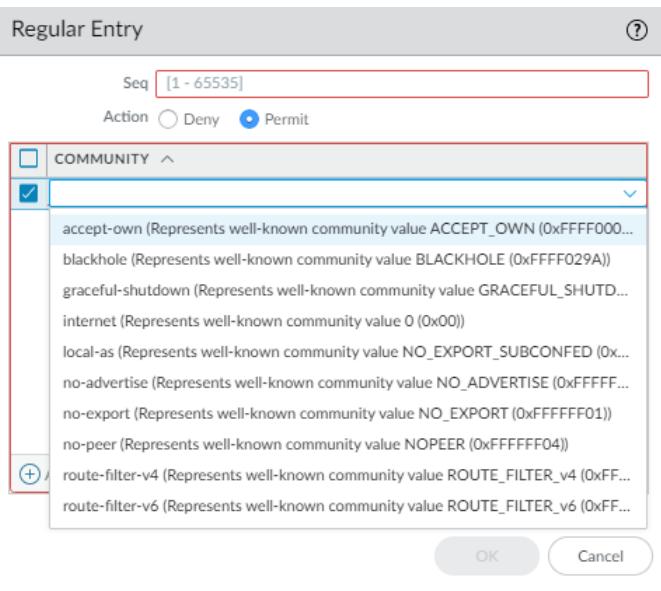
- **Regular**—Add a **Seq** number (range is 1 to 65,535), select the **Action: Deny** (the default) or **Permit**, and **Add** one or more community values, select one or more well-known communities, or enter a combination of community values and well-known communities. Separate multiple communities with a vertical bar (|), for example,

6409:10|6520:13|internet. Enter a maximum of 16 communities in a **Regular** entry (rule).

- A regular community value in the format AA:NN where AA is an AS number and NN is a network number (each with a range of 0 to 65,535).
- **accept-own**—Represents well-known community value ACCEPT-OWN (0xFFFF0001)
- **blackhole**—Represents well-known community value BLACKHOLE (0xFFFF029A). The neighboring network should discard traffic destined for the prefix.
- **graceful-shutdown**—Represents well-known community value GRACEFUL_SHUTDOWN (0xFFFF0000)
- **internet**—Represents well-known community value 0 (0x00). Advertise a prefix to all BGP neighbors.
- **local-as**—Represents well-known community value NO_EXPORT_SUBCONFED (0xFFFFFFF03). The effect is to not advertise the prefix outside of the sub-AS in a confederation.
- **no-advertise**—Represents well-known community value NO_ADVERTISE (0xFFFFFFF02). Adding this community to a prefix means the receiving BGP peer

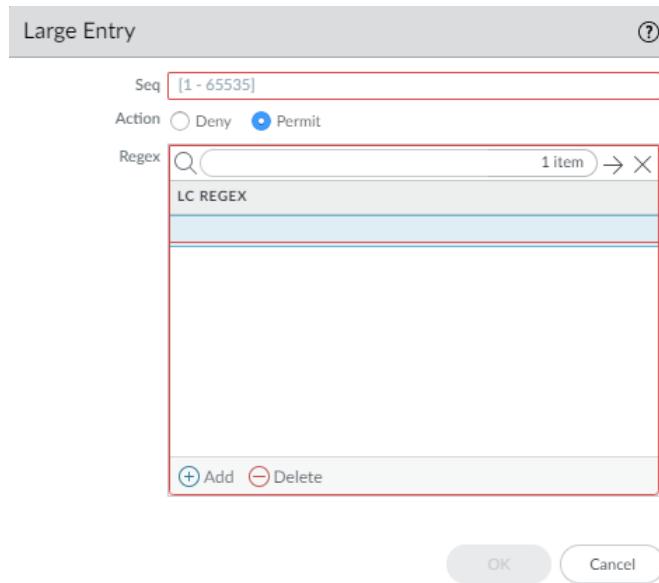
will place the prefix in its BGP route table, but won't advertise the prefix to other neighbors.

- **no-export**—Represents well-known community value NO_EXPORT (0xFFFFFFF01). Adding this community to a prefix means the receiving BGP peer will advertise the prefix only to iBGP neighbors, not neighbors outside the AS.
- **no-peer**—Represents well-known community value NOPEER (0xFFFFFFF04).
- **route-filter-v4**—Represents well-known community value ROUTE_FILTER_v4 (0xFFFF0003).
- **route-filter-v6**—Represents well-known community value ROUTE_FILTER_v6 (0xFFFF0005).

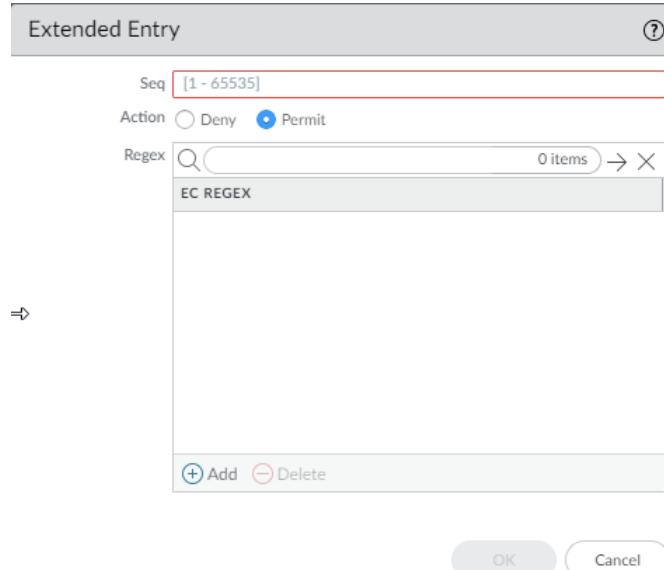


- **Large**—Add a **Seq** number (range is 1 to 65,535), select the **Action: Deny** (the default) or **Permit**, and **Add** a large community regular expression (LC REGEX) entry. Characters allowed in an entry are 1234567890_^.{|}{()}\$^*+.?-\. Each community must be in the format **regex1:regex2:regex3**; for example,

203[1-2]:205[2-5]:206[5-6]. Enter a maximum of eight communities in a Large entry (rule).



- **Extended—Add a Seq number (range is 1 to 65,535), select the Action: Deny (the default) or Permit, and Add the BGP extended community regular expression (EC REGEX). Characters allowed are 1234567890_^.{}\$^*+.?-\. Each extended community must be in the format **regex1:regex2**; for example, 204*[3-8]:205*[4-8]. Enter a maximum of eight communities in an Extended entry (rule).**



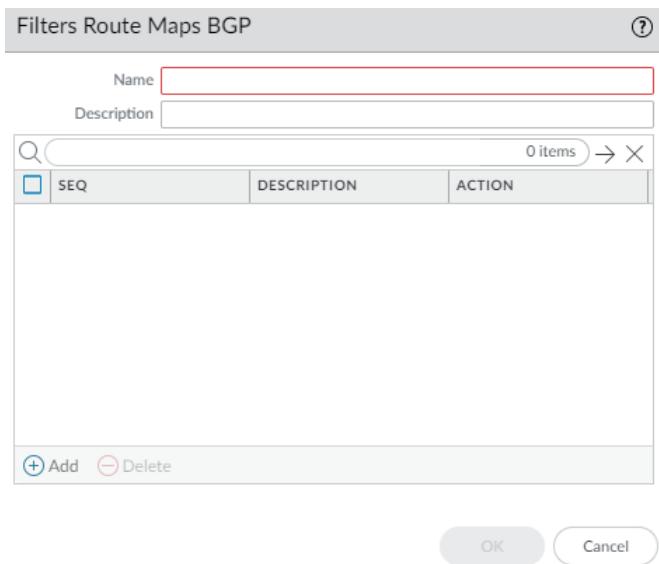
5. Click **OK** to save the entry in the Community List. Optionally add more entries of the same type (Regular, Large, or Extended).
6. Click **OK** to save the Community List.

STEP 5 | Create a BGP route map.

1. Select **Network > Routing > Routing Profiles > Filters**.
2. Add a **Filters Route Maps BGP by Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a

combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.

3. Enter a helpful **Description** of the route map.

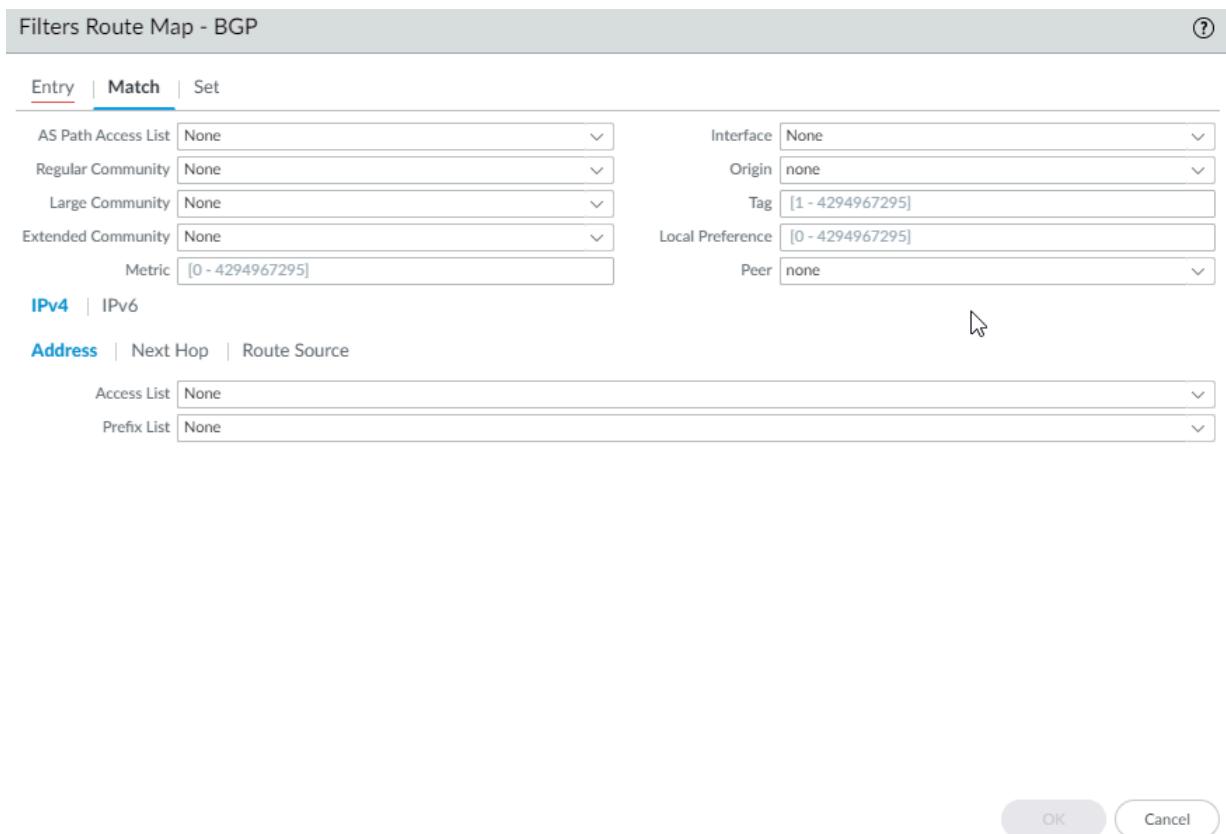


4. Add a route map and on the **Entry** tab, assign a **Seq** number; range is 1 to 65,535.



Assign sequence numbers that are five or more numbers apart so you have unused numbers for which to insert additional entries in the future.

5. Enter a helpful **Description** of the entry (rule).
6. For **Action**, select **Deny** or **Permit**.
7. On the **Match** tab, specify the criteria that determine which routes are subject to the function that uses this route map. Multiple attributes are logically ANDed, meaning all criteria must be met.



- **AS Path Access List**—Select an AS path list. Default is **None**.
- **Regular Community**—Select a Community list. Default is **None**.
- **Large Community**—Select a Large Community list. Default is **None**.
- **Extended Community**—Select an Extended Community list. Default is **None**.
- **Metric**—Enter a value in the range 0 to 4,294,967,295.
- **Interface**—Select a local interface from the list of all interfaces for all logical routers. Make sure to choose an interface that belongs to the logical router you are

configuring. Default is **None**. At commit, the firewall checks that the interface you chose belongs to the logical router you are configuring.

- **Origin**—Select the origin of the route: **ebgp**, **ibgp**, or **incomplete**. Default is **none**.
 - **Tag**—Enter a tag value that has meaning in your networks, in the range 0 to 4,294,967,295.
 - **Local Preference**—Enter a value in the range 0 to 4,294,967,295.
 - **Peer**—Select a peer name or **local (Static or Redistributed routes)**. Default is **none**.
8. Select **IPv4** or **IPv6** to match on various types of addresses. If you select **IPv4**:
- On the **Address** tab, select an **Access List** to specify addresses to match.
 - Select a **Prefix List** to specify addresses to match. It matches the prefix received from a peer or a prefix redistributed to protocol from another protocol.
 -  *If both an access list and prefix list are specified, both requirements must be met (logical AND).*
 - On the **Next Hop** tab, select an **Access List** to specify next hop addresses to match.
 - Select a **Prefix List** to specify next hop addresses to match.
 - On the **Route Source** tab, select an **Access List** to specify a source IP address of a route to match. For example, the access list could permit a distant peer with the address 192.168.2.2 who is advertising a route to a certain prefix. You can make this BGP route map match on the route's source address 192.168.2.2 and then perhaps

filter the route based on matching the peer address 192.168.2.2 as the source of the route, or set a next hop for routes matching that route source.

- Specify a **Prefix List** to specify one or more source network prefixes to match.
- If you select **IPv6**:
 - On the **Address** tab, select an **Access List** to specify addresses to match.
 - Select a **Prefix List** to specify addresses to match.
 - On the **Next Hop** tab, select an **Access List** to specify next hop addresses to match.

- Set** any of the following attributes for routes that meet the match criteria:

The screenshot shows the 'Filters Route Map - BGP' configuration dialog. The 'Set' tab is active. The interface includes sections for 'Aggregator' (with 'Enable BGP atomic aggregate' checked), 'IP' (IPv4 selected), 'AS Path' (with 'ASPATH EXCLUDE' and 'ASPATH PREPEND' sections), and 'Community' (with 'Regular Community' and 'Large Community' sections). At the bottom are 'OK' and 'Cancel' buttons.

- Enable BGP atomic aggregate**—Mark the route as a less specific route because it has been aggregated. ATOMIC_AGGREGATE is a well-known discretionary attribute that alerts BGP speakers along a path that information has been lost due to route aggregation, and therefore the aggregate path might not be the best path to the destination. When some router are aggregated by an aggregator, the aggregator attaches its Router-ID to the aggregated route into the AGGREGATOR-ID attribute

and it sets the ATOMIC_AGGREGATE attribute or not, based on whether the AS_PATH information from the aggregated routers was preserved.

- **Aggregator AS**—Enter the Aggregator AS. The Aggregator attribute includes the AS number and the IP address of the router that originated the aggregated route. The IP address is the Router ID of the router that performs the route aggregation.
- **Router ID**—Enter the aggregator's Router ID (usually a loopback address).
- **Local Preference**—Enter the local preference to which matching routes are set; range is 0 to 4,294,967,295. IBGP Update packets carry local preference, which is advertised to IBGP peers only. When there are multiple routes to another AS, the firewall prefers the highest local preference.
- **Tag**—Set a tag; range is 1 to 4,294,967,295.
- **Metric Action**—Select an action: **set**, **add**, or **subtract**. You can set the specified Metric Value, or add the specified Metric Value to the matching route's original metric value, or subtract the specified Metric Value from the matching route's original metric value; default is **set**. Select the **add** or **subtract** action to adjust a metric and thus prioritize or deprioritize the matching route.
- **Metric Value**—Enter the metric value to set matching routes to, or add to, or subtract from the original metric value; range is 0 to 4,294,967,295.
- **Weight**—Set a weight (applied locally; not propagated); range is 0 to 4,294,967,295.
- **Origin**—Set the origin of the matching routes: **ebgp**, **ibgp**, or **incomplete** (unclear how the route came to be added to the RIB).
- **Originator ID**—Set the IP address of the originator of the matching routes.
- **Delete Regular Community**—Select a regular community to delete. Default is **None**.
- **Delete Large Community**—Select a large community to delete. Default is **None**.
- Select **IPv4** or **IPv6** as the AFI.
- On the **IPv4** tab, select a **Source Address** to set from the list of all source addresses from all logical routers or select **None**. At commit, the firewall checks that the source address you chose belongs to the logical router you are configuring.
- Select an **IPv4 Next-Hop** to set: **none**, **peer-address (Use Peer Address)**, or **unchanged**.
- On the **IPv6** tab, select **IPv6 Nexthop Prefer Global Address** to prefer the global unicast address over the other IPv6 address types (link-local address, anycast address,

or multicast address) for next hop. (By default, connected peers prefer a link-local next hop address over a global next hop address.)

- On the **IPv6** tab, select a **Source Address** to set from the list of all source addresses from all logical routers or select **None**. At commit, the firewall checks that the source address you chose belongs to the logical router you are configuring.
- Select an **IPv6 Next-Hop** to set: **none** or **peer-address (Use Peer Address)**.
- In the AS Path window, **Add** up to four AS paths to **Exclude** from the AS path of matching routes, perhaps to remove an AS from a confederation.
- **Add** up to four AS Paths to **Prepend** to the AS Path of matching route(s) (to make the route in an advertisement less desirable).
- In the Regular Community window, select **Overwrite Regular Community** to overwrite the regular community.
- **Add a Regular Community** to add one or more regular communities.
- In the Large Community window, select **Overwrite Large Community** to overwrite the large community.
- **Add a Large Community** to add one or more large communities.
- In the Regular Community window, select **Overwrite Regular Community** to overwrite the regular community.
- **Add a Regular Community** to add one or more regular communities.
- In the Large Community window, select **Overwrite Large Community** to overwrite the large community.
- **Add a Large Community** to add one or more large communities.

11. Click **OK** to save the route map entry. Optionally add more entries.

12. Click **OK** to save the BGP route map.

STEP 6 | Create a Redistribution Route Map.

1. Select **Network > Routing > Routing Profiles > Filters**.
2. **Add a Filters Route Maps Redistribution by Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Enter a helpful **Description**.
4. To redistribute from a **Source Protocol**, select **BGP**, **OSPF**, **OSPFv3**, **RIP**, or **Connected Static**. The source protocol is where the Match selections apply.
5. To redistribute the routes to a **Destination Protocol** or local RIB, select **BGP**, **OSPF**, **OSPFv3**, **RIP**, or **Rib**. The destination protocol is where the Set selections apply. The

Destination Protocols available in the dropdown depend on the Source Protocol selected. (This step shows an example of BGP redistributed to OSPF.)

Filters Route Maps Redistribution

Name <input type="text"/>		
Description <input type="text"/>		
Source Protocol BGP		
Destination Protocol OSPF		
SEQ	DESCRIPTION	ACTION
+ Add Delete		

OK Cancel

6. **Add an Entry** and enter the **Seq** number (range is 1 to 65,535).
7. Enter a helpful **Description**.
8. Select the **Action: Deny or Permit**.
9. Select the **Match** tab to configure criteria for the source protocol; this example specifies BGP attributes to match.

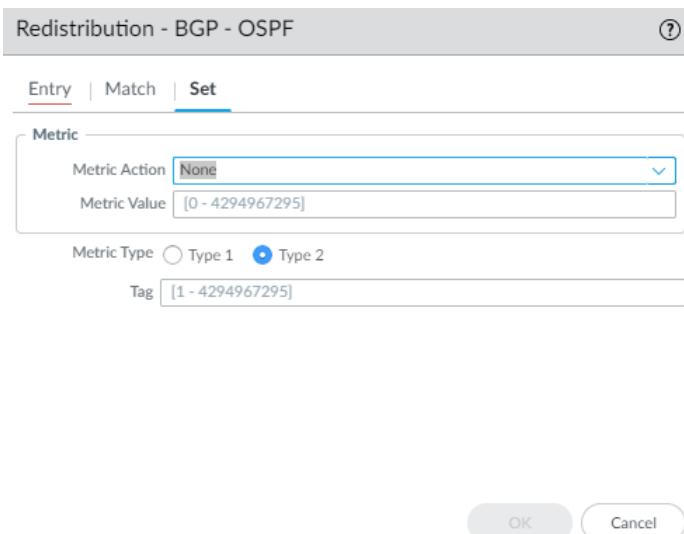
Redistribution - BGP - OSPF

Entry	Match	Set
AS Path Access List <input type="text"/> None	Interface <input type="text"/> None	
Regular Community <input type="text"/> None	Origin <input type="text"/> none	
Large Community <input type="text"/> None	Tag <input type="text"/> [1 - 4294967295]	
Extended Community <input type="text"/> None	Local Preference <input type="text"/> [0 - 4294967295]	
Metric <input type="text"/> [0 - 4294967295]	Peer <input type="text"/> none	
Address Next Hop Route Source		
Access List <input type="text"/> None		
Prefix List <input type="text"/> None		

OK Cancel

10. Select an **AS Path Access List**; default is **None**.
11. Select a **Regular Community**; default is **None**.
12. Select a **Large Community**; default is **None**.
13. Select an **Extended Community**; default is **None**.
14. Enter a **Metric**; range is 0 to 4,294,967,295.

15. Select an **Interface**; default is **None**.
16. Select the **Origin** of the route: **ebgp**, **ibgp**, or **incomplete**; default is **none**.
17. Enter a **Tag**; range is 1 to 4,294,967,295.
18. Enter a **Local Preference**; range is 0 to 4,294,967,295.
19. Select a **Peer name or local (Static or Redistributed routes)**; default is **none**.
20. The **Address** tab refers to the Destination address in a route. Select an **Access List** to specify routes with a destination address that must match in order to be redistributed. Default is **None**.
21. Select a **Prefix List** to specify routes with a destination address that must match in order to be redistributed. Default is **None**.
22. Select the **Set** tab to configure actions to perform on routes matching this rule, which will be redistributed to the destination protocol. (In this example, the destination protocol is OSPF.)



23. Select the **Metric Action** for the redistribution rule: you can **set** the Metric value, **add** the specified **Metric Value** to the matching route's original Metric value, or **subtract** the specified **Metric Value** from the matching route's original Metric value; default is **None**. Select the **add** or **subtract** action to adjust a metric and thus prioritize or deprioritize the matching route.

For example, you can put the metric of an IGP into BGP by using redistribution. The metric is dynamic, and you can simply add to its value rather than set it to an absolute number.

24. Enter a **Metric Value** to set, add to, or subtract from the metric; range is 0 to 4,294,967,295.
25. Select the **Metric Type**: **Type 1** or **Type 2** (because this example uses OSPF as the destination protocol).
26. Specify a **Tag**; range is 1 to 4,294,967,295.
27. Click **OK** to save the rule. Optionally add more rules.
28. Click **OK** to save the Redistribution route map.

Configure OSPFv2 on an Advanced Routing Engine

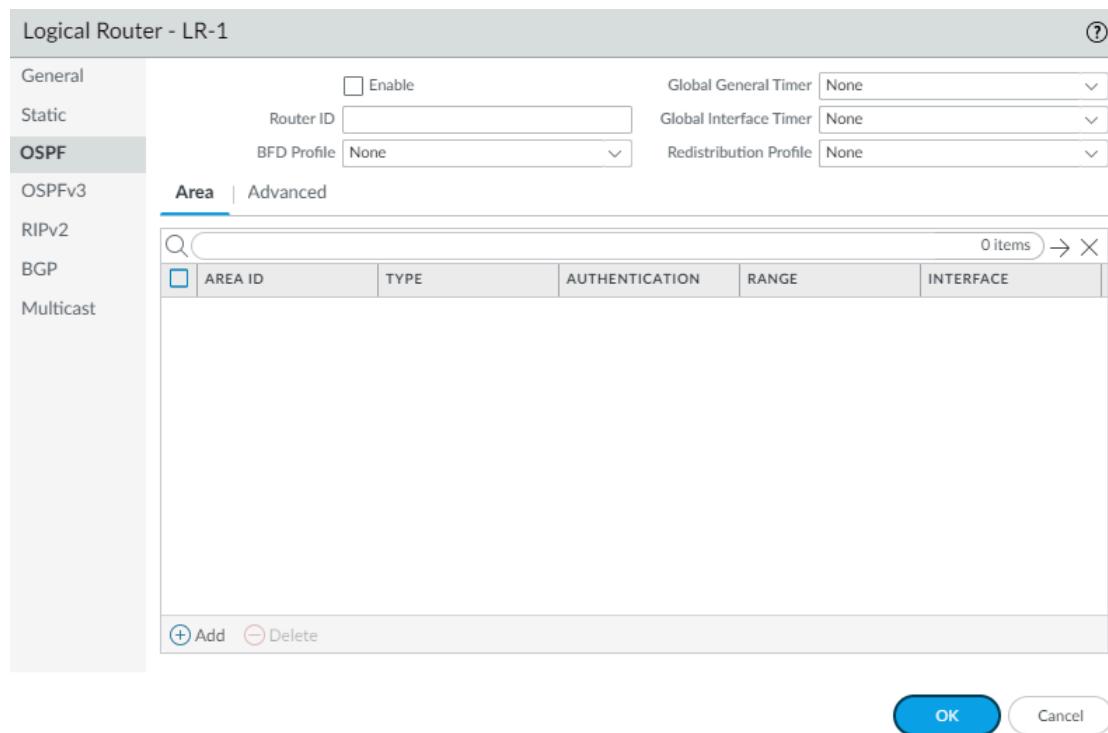
The Advanced Routing Engine supports OSPFv2, which supports only IPv4 addressing. Before you configure OSPFv2, you should understand [OSPF Concepts](#).

Consider the [OSPF Routing Profiles](#) and [filters](#) that you can apply to OSPF and thereby save configuration time and maintain consistency. You can create profiles and filters in advance or as you configure OSPFv2.

STEP 1 | Configure a Logical Router.

STEP 2 | Enable OSPFv2 and configure general settings.

1. Select **Network > Routing > Logical Routers** and select a logical router.
2. Select **OSPF** and **Enable it**.



3. Enter the **Router ID** in the format of an IPv4 address.
4. If you want to apply BFD to OSPF, select a **BFD Profile** you created, or select the **default** profile, or [create a new BFD Profile](#). Default is **None (Disable BFD)**.
5. Select an **OSPF Global General Timer** profile or [create a new one](#).
6. Select an **OSPF Global Interface Timer** profile or [create a new one](#).
7. Select an **OSPF Redistribution Profile** or [create a new one](#) to redistribute IPv4 static routes, connected routes, RIPv2 routes, IPv4 BGP routes, or the IPv4 default route to OSPF.

STEP 3 | Create an OSPF area and specify characteristics based on the type of area.

1. Select **Area** and **Add** an area identified by its **Area ID** in x.x.x.x format. This is the identifier that each neighbor must accept to be part of the same area.
2. Select the **Type** tab and for **Authentication**, select an Authentication profile or [create a new Authentication profile](#).
3. Select the **Type** of area:
 - **Normal**—There are no restrictions; the area can carry all types of routes (intra-area routes, inter-area routes, and external routes).
 - **Stub**—There is no outlet from the area. To reach a destination outside of the area, traffic must go through an Area Border Router (ABR), which connects to other areas.
 - **NSSA (Not-So-Stubby-Area)**—NSSAs implement stub or totally stubby functionality, yet contain an autonomous system boundary router (ASBR). Type 7 LSAs generated by the ASBR are converted to Type 5 by ABRs and flooded to the rest of the OSPF domain. (The next graphic shows NSSA selected.)
4. (**Stub and NSSA areas only**) Select **no-summary** to prevent the area from receiving Type 3 Summary LSAs and thereby reduce traffic in the area.
5. (**NSSA area only**) Select **Default information originate** to cause OSPF to originate a default route.
 - Enter a **Metric** for the default route; range is 1 to 16,777,214; default is 10.
 - Select the **Metric-Type: Type 1 or Type 2**. Type E1 cost is the sum of the external cost plus the internal cost to reach that route. Type E2 is only the external cost of

that route. This can be useful when you want to load-balance the same external route, for example.

The screenshot shows the 'OSPF - Area' configuration dialog. The 'Area ID' field is empty. The 'Type' tab is selected, showing 'Authentication: None' and 'Type: NSSA'. Under 'Default information originate', there is a checked checkbox for 'no-summary' and a checked checkbox for 'Default information originate'. The 'Metric' is set to 10 and 'Metric-Type' is set to 'Type 1'. Under 'ABR', there are four dropdown menus: 'Import-list' (None), 'Export-list' (None), 'Inbound Filter List' (None), and 'Outbound Filter List' (None). To the right of these dropdowns is a table titled 'IPV4 PREFIX' with a search bar, a button to add items ('0 items'), and a delete button. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

6. Select **ABR** to filter prefixes going in or out of the area, and then configure the following filters:
 - Select an **Import-list** or [create a new Access List](#) to filter network routes coming from another router into the area in LSAs, based on IPv4 source address, thus allowing

or preventing the routes from being added to the global RIB (leave the destination address of the access list empty).

- Select an **Export-list** or [create a new Access List](#) to filter network routes that originated in the area, to allow or prevent the routes from being advertised to other areas.
- Select an **Inbound Filter List** or [create a new Prefix List](#) to filter network prefixes coming into the area.
- Select an **Outbound Filter List** or [create a new Prefix List](#) to filter network prefixes that originated in the area, to prevent the routes from being advertised to other areas.
- If the **Type** of area is **NSSA** and **ABR** is selected, **Add an IPv4 Prefix** to summarize a group of external subnets into a single Type-7 LSA, which is then translated to a Type-5 LSA and advertised to the backbone when you select **Advertise**.

STEP 4 | Specify the network range for the area.

1. Select **Range** and **Add an IP Address/Netmask**, which summarizes routes for the area. The result is that a Type-3 Summary LSA with routing information matching this range

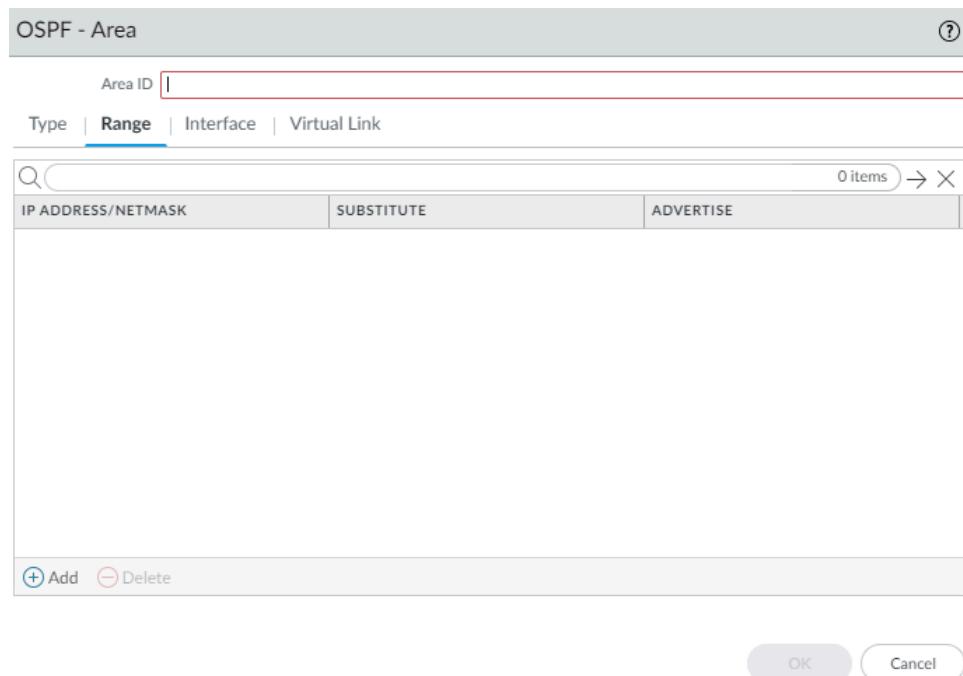
is advertised into the backbone area if that area contains at least one intra-area network (that is, described with router or network LSA) from this range.



Look at the learned routes in the LSDB for the area and use this Range to summarize routes, thereby reducing LSA traffic.

2. Enter a **Substitute** IP address/netmask so that a Type-3 Summary LSA with this IP address/netmask is announced into the backbone area if the area contains at least one intra-area network from the **IP Address/Netmask** specified in the prior step.

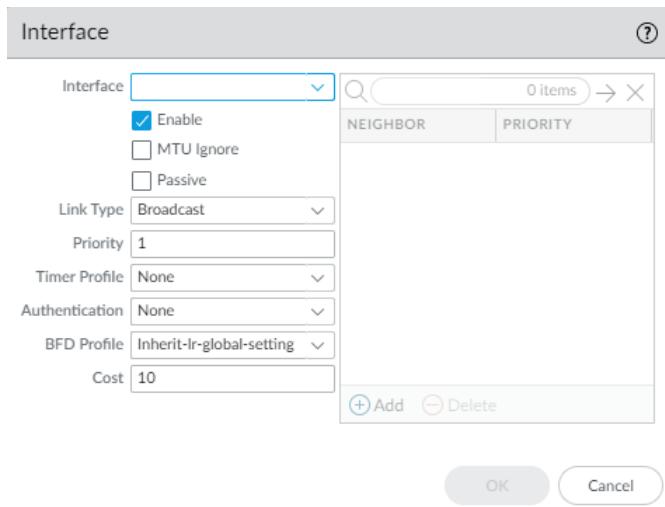
Use the Substitute IP address/netmask as a way to translate a private address to a public address. A Substitute address has no effect if Advertise is disabled.
3. Select **Advertise** to send link-state advertisements (LSAs) that match the subnet; default is enabled.



The screenshot shows the 'OSPF - Area' configuration window. At the top, there's a search bar and a help icon. Below it, the 'Area ID' field is empty. The navigation tabs include 'Type', 'Range' (which is selected), 'Interface', and 'Virtual Link'. Under the 'Range' tab, there's a table with three columns: 'IP ADDRESS/NETMASK', 'SUBSTITUTE', and 'ADVERTISE'. The table header shows '0 items'. At the bottom of the table, there are 'Add' and 'Delete' buttons. At the very bottom of the window are 'OK' and 'Cancel' buttons.

STEP 5 | Configure each interface to be included in the area.

1. **Add an Interface** by selecting one and **Enable** it.
2. Select **MTU Ignore** to ignore maximum transmission unit (MTU) mismatches when trying to establish an adjacency (default is disabled; MTU match checking occurs). [RFC 2328](#)

- defines the interface MTU as “The size in bytes of the largest IP datagram that can be sent out the associated interface, without fragmentation.”
3. Select **Passive** to allow the network of the interface to be advertised, but no neighbor relationship is established on that interface; this is useful for leaf interfaces.
- 
4. Select the **Link Type**:
- **Broadcast**—All neighbors that are accessible through the interface are discovered automatically by multicasting OSPF Hello messages, such as over an Ethernet interface.
 - **p2p** (point-to-point)—Automatically discover the neighbor.
 - **p2mp** (point-to-multipoint)—Neighbors must be defined manually: **Add** the Neighbor IP address for all neighbors that are reachable through this interface and the **Priority** of each neighbor to be elected the designated router (DR) or backup DR; range is 0 to 255; default is 1.
5. Enter the **OSPF Priority** for the interface to be elected as a designated router (DR) or backup DR (BDR); range is 0 to 255; default is 1. If zero is configured, the router will not be elected as DR or BDR.
6. Select a **Timer Profile** to apply to the interface or [create a new OSPF Interface Timer profile](#). This OSPF Interface Timer profile overrides the Global Interface Timer applied to OSPF.
7. Select an **Authentication Profile** to apply to the interface or [create a new OSPF Interface Authentication profile](#). This Authentication Profile overrides the Authentication Profile applied to the Area (on the Type tab).
8. By default, the interface will inherit the BFD profile you applied to the logical router for OSPF (**Inherit-Ir-global-setting**). Alternatively, select the **default** profile, select a different **BFD Profile**, [create a new BFD Profile](#), or select **None (Disable BFD)** to disable BFD for the interface.
9. Enter an **OSPF Cost** for the interface, which influences route selection; range is 1 to 65,535; default is 10. During route selection, a route with a lower cumulative cost (the added costs of each interface used) is preferred over a route with a higher cumulative cost.
10. Click **OK**.

STEP 6 | If the ABR does not have a physical link to the backbone area, configure a virtual link to a neighbor ABR within the same area that has a physical link to the backbone area.

1. Select **Virtual Link**.
2. Add a virtual link by **Name**.
3. **Enable** the virtual link.

The screenshot shows the 'OSPF - Area - Virtual Link' configuration dialog. It includes fields for Name (with a red border), Enable (checkbox checked), Area (dropdown menu), Router ID (text input), Timer Profile (dropdown menu with 'None'), and Authentication (dropdown menu with 'None'). At the bottom are 'OK' and 'Cancel' buttons.

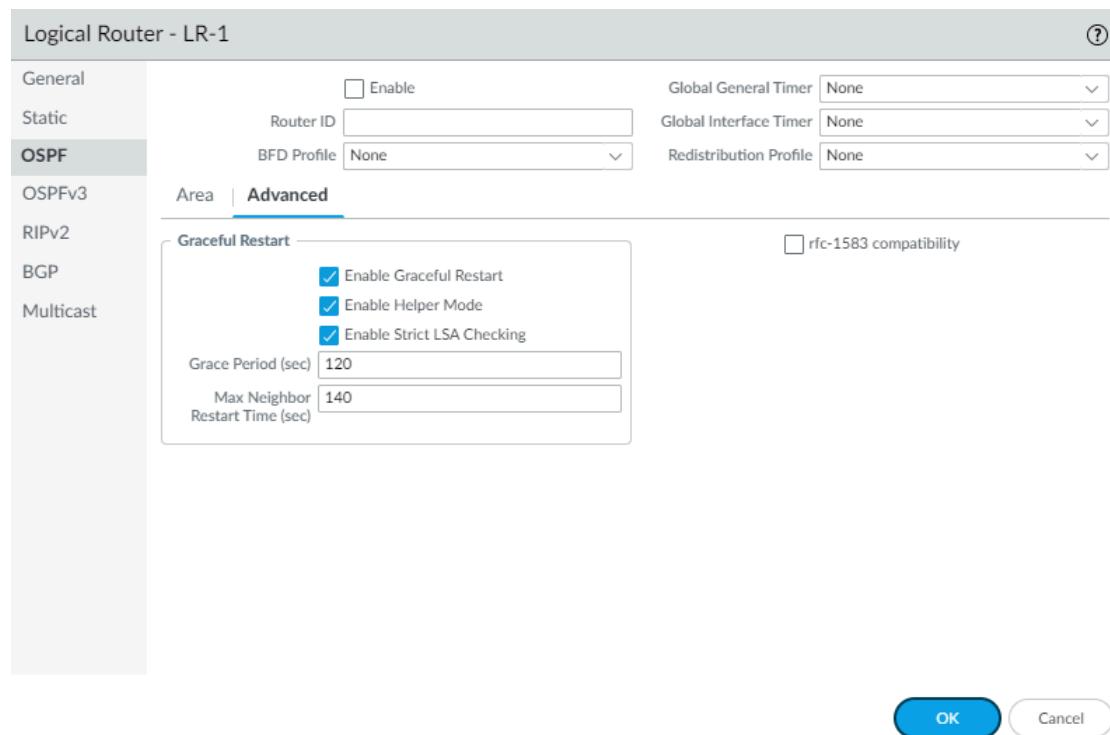
4. Select the transit **Area** where the neighbor ABR that has the physical link to the backbone area is located.
5. Enter the **Router ID** of the neighbor ABR on the remote end of the virtual link.
6. Select a **Timer Profile** or [create a new Timer Profile](#) to apply to the virtual link. This OSPF Interface Timer profile overrides the Global Interface Timer applied to OSPF and the OSPF Interface Timer profile applied to the interface.
7. Select an **Authentication** profile or [create a new Authentication Profile](#) to apply to the virtual link. This Authentication Profile overrides the Authentication Profile applied to the Area (on the Type tab) and the Authentication Profile applied to the interface.
8. Click **OK**.

STEP 7 | Click **OK** to save the area.

STEP 8 | Configure [OSPF Graceful Restart](#) and [RFC 1583](#) compatibility for OSPFv2.

1. Select **Network > Routing > Logical Routers** and select the logical router.
2. Select **OSPF > Advanced**.
3. Select **rfc-1583 compatibility** to enforce compatibility with RFC 1583, which allows one best route to an autonomous system boundary router (ASBR) in the OSPF routing table.

Default is disabled, which means the OSPF routing table can maintain multiple intra-AS paths in the routing table, thereby preventing routing loops.

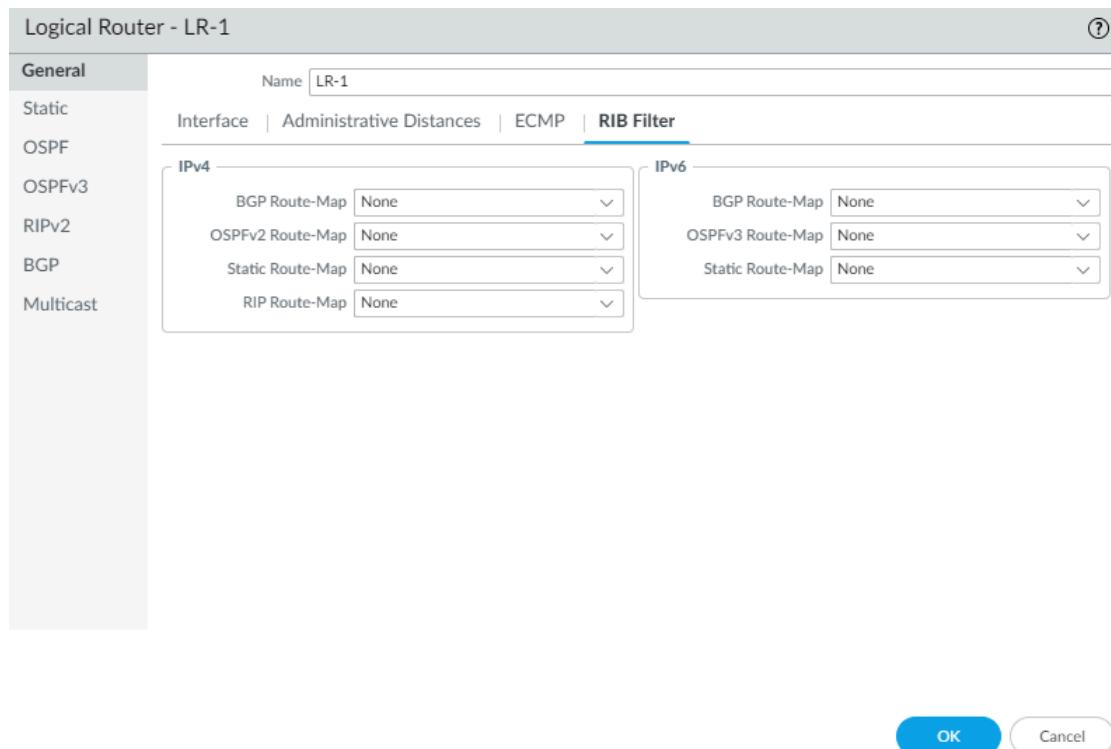


4. **Enable Graceful Restart** to enable [OSPF Graceful Restart](#) for the logical router. Default is enabled.
5. **Enable Helper Mode** to enable the logical router to function in Graceful Restart helper mode. Default is enabled.
6. **Enable Strict LSA Checking** to cause the helper router to stop performing helper mode and causes the graceful restart process to stop if a link-state advertisement indicates a network topology change. Default is enabled.
7. Specify the **Grace Period (sec)**—the number of seconds within which the logical router will perform a graceful restart if the firewall goes down or becomes unavailable; range is 5 to 1,800; default is 120.
8. Specify the **Max Neighbor Restart Time (sec)**; range is 5 to 1,800; default is 140.
9. Click **OK**.

STEP 9 | Configure intra-area filtering to determine which OSPFv2 routes are placed in the global RIB.

You might learn OSPFv2 routes and redistribute them, but not want them in the global RIB; you might want to allow only specific OSPFv2 routes to the global RIB.

1. Select **Network > Routing > Logical Routers** and select a logical router.
2. Select **RIB Filter**.
3. To filter IPv4 OSPFv2 routes for the global RIB, in the **OSPFv2 Route-Map**, select a Redistribution route map you created or [create a new Redistribution Route Map](#) in which the Source Protocol is OSPF and the Destination Protocol is RIB.



4. Click **OK**.

STEP 10 | (Optional) Change the default administrative distances for OSPF intra area, inter area, and external routes [within a logical router](#).

STEP 11 | Commit.

STEP 12 | View advanced routing information for OSPFv2 and the link-state database (LSDB). The PAN-OS CLI Quick Start lists the commands in the [CLI Cheat Sheet: Networking](#).

Create OSPF Routing Profiles

The Advanced Routing Engine supports OSPFv2; create the following profiles to apply to the protocol, making the configuration easier and more consistent. The profiles can be used across multiple logical routers and virtual systems. This topic describes the profiles and how to configure them.

- **OSPF Global Timer Profiles**—Configure the timer for link-state advertisement (LSA) min-arrival and Shortest Path First (SPF) timers for OSPFv2 areas. Apply the profile in the OSPF general configuration.
- **OSPF Interface Authentication Profiles**—Specify authentication using a password or MD5; apply such profiles to an OSPF area, an interface, and/or a virtual link.
- **OSPF Interface Timer Profiles**—Configure timers related to interface operations, such as OSPF hello and graceful restart. Apply such profiles to the OSPF general configuration, an interface, and/or a virtual link.
- **OSPF Redistribution Profiles**—Specify how to redistribute IPv4 static routes, connected routes, BGP IPv4 routes, RIPv2 routes, and the IPv4 default route to OSPF. Apply the profile in the OSPF general configuration.

STEP 1 | Create an OSPF Global Timer Profile.

1. Select **Network > Routing > Routing Profiles > OSPF**.
2. Add an **OSPF Global Timer Profile** by **Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Enter the **LSA min-arrival**, which is the minimum length time (in seconds) between transmissions of two instances of the same LSA (same advertising router ID, same LSA type, and same LSA ID). If the same LSA arrives sooner than the configured interval, the LSA is dropped. Range is 1 to 10; default is 5. The LSA min-arrival is equivalent to MinLSInterval in RFC 2328. Lower values can be used to reduce re-convergence times when topology changes occur.
4. In the SPF area, enter the **Initial delay** (in seconds) from when the logical router receives a topology change until it performs the Shortest Path First (SPF) calculation; range is 0 to

600; default is 5. Lower values enable faster OSPF re-convergence. Routers peering with the firewall should use the same delay value to optimize convergence times.

5. Enter the **Initial hold time** (in seconds) between consecutive SPF calculations; range is 0 to 600; default is 5.
6. Enter the **Maximum hold time** (in seconds), which is the largest value that the hold time throttles to until remaining steady; range is 0 to 600; default is 5.

The dialog box is titled "OSPF Global Timer Profile". It has a "Name" field (highlighted with a red border). Under the "Throttle" section, there is a "LSA min-arrival" field containing the value "5". Under the "SPF" section, there are three fields: "Initial delay" (value "5"), "Initial hold time" (value "5"), and "Maximum hold time" (value "5"). At the bottom are "OK" and "Cancel" buttons.

7. Click **OK**.

STEP 2 | Create an OSPF Interface Authentication Profile.

1. Select **Network > Routing > Routing Profiles > OSPF**.
2. **Add an OSPF Auth Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Select the **Type** of authentication: **Password** or **MD5**.
 - If you choose **Password**, enter the **Password** (maximum of eight characters) and **Confirm Password**.

The dialog box is titled "OSPF Auth Profile". It has a "Name" field (highlighted with a red border). Below it is a "Type" section with two radio buttons: "Password" (selected) and "MD5". There are two password input fields: "Password" and "Confirm Password". At the bottom are "OK" and "Cancel" buttons.

- If you choose **MD5**, **Add an MD5 key ID** (range is 0 to 255) and a **Key** (a maximum of 16 alphanumeric characters). Select **Preferred** to prefer an MD5 key over other MD5 keys. During the commit, the firewall goes through the list of keys from the top down and the Preferred key is moved to the top of the list; the top Preferred key is used. (In

other words, if you select more than one Preferred MD5 key, the last one chosen as Preferred is the Preferred key.)

The screenshot shows the 'OSPF Auth Profile' dialog box. At the top, there is a 'Name' input field with a red border, indicating it is required. Below it, a 'Type' section has two radio buttons: 'Password' (unchecked) and 'MD5' (checked). A search bar with a magnifying glass icon and a '0 items' count are also present. The main area is a table with three columns: 'MD5', 'KEY', and 'PREFERRED'. The 'PREFERRED' column contains a single entry 'MD5'. At the bottom, there are 'Add' (+) and 'Delete' (-) buttons, and 'OK' and 'Cancel' buttons.

4. Click **OK**.

STEP 3 | Create an OSPF Interface Timer Profile.

1. Select **Network > Routing > Routing Profiles > OSPF**.
2. **Add an OSPF Interface Timer Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain

a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.

The dialog box is titled "OSPF Interface Timer Profile". It contains the following fields:

Name	<input type="text"/>
Hello Interval	10
Dead Count	4
Retransmit Interval	5
Transmit Delay	1
Graceful Restart Hello Delay (sec)	10

Buttons at the bottom: OK, Cancel.

3. Enter the **Hello Interval**, the interval (in seconds) between Hello packets that the firewall sends out an interface to maintain neighbor relationships; range is 1 to 3600; default is 10.
4. Enter the **Dead Count**, the number of times the Hello Interval can occur for a neighbor without OSPF receiving a hello packet from the neighbor, before OSPF considers that neighbor down; range is 3 to 20; default is 4.
5. Enter the **Retransmit Interval**, the number of seconds between LSA retransmissions to adjacent routers; range is 1 to 1800; default is 5.
6. Enter the **Transmit Delay**, the number of seconds required to transmit a Link State Update Packet over the interface. Link State Advertisements in the update packet have their age incremented by this number before they are transmitted; range is 1 to 1800; default is 1.
7. Enter the **Graceful Restart Hello Delay (sec)** in seconds, which applies to an OSPF interface when Active/Passive High Availability is configured. Graceful Restart Hello Delay is the length of time during which the firewall sends Grace LSA packets at 1-second intervals. During this time, no hello packets are sent from the restarting firewall. During the restart, the dead timer (which is the Hello Interval multiplied by the Dead Count) is also counting down. If the dead timer is too short, the adjacency will go down during the graceful restart because of the hello delay. Therefore, it is recommended that the dead timer be at least four times the value of the Graceful Restart Hello Delay. For example, a Hello Interval of 10 seconds and a Dead Count of 4 yield a dead timer of 40 seconds. If the Graceful Restart Hello Delay is set to 10 seconds, that 10-second delay of hello packets is comfortably within the 40-second dead timer, so the adjacency will not time out during a graceful restart. Range is 1 to 10; default is 10.
8. Click **OK**.

STEP 4 | Create an OSPF Redistribution Profile to specify any combination of IPv4 static routes, connected routes, BGP IPv4 routes, RIPv2 routes, and default IPv4 route to redistribute to OSPF.

1. Select **Network > Routing > Routing Profiles > OSPF**.
2. Add an **OSPF Redistribution Profile** by **Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain

a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.

The screenshot shows the 'OSPF Redistribution Profile' dialog box. At the top, there is a 'Name' field with a red border. Below it are four sections: 'IPv4 Static', 'Connected', 'RIPv2', and 'BGP AFI IPv4'. Each section has an 'Enable' checkbox (checked for all), a 'Metric' input field (ranging from 1 to 65,535), a 'Metric-Type' radio button group (Type 1 or Type 2, with Type 2 selected for most sections), and a 'Redistribute Route-Map' dropdown menu (set to 'None'). The 'BGP AFI IPv4' section also includes 'Always' and 'Enable' checkboxes. At the bottom right are 'OK' and 'Cancel' buttons.

3. Select **IPv4 Static** to allow configuration of this portion of the profile.
 - Enable the IPv4 Static portion of the profile.
 - Specify the **Metric** to apply to the static routes being redistributed into OSPF (range is 1 to 65,535).
 - Specify the **Metric-Type**: **Type 1** (OSPF costs) or **Type 2** (default). If there are two static routes to a destination and they have the same cost, a Type 2 route is preferred over a Type 1 route.
 - Select a **Redistribute Route-Map** or [create a new Redistribute Route Map](#) whose match criteria control which IPv4 static routes to redistribute into OSPF. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route. Likewise, the Metric Type in the route map Set configuration takes precedence over the Metric Type configured in this redistribution profile.
4. Select **Connected** to allow configuration of this portion of the profile.
 - Enable the Connected portion of the profile.
 - Specify the **Metric** to apply to the connected routes being redistributed into OSPF (range is 1 to 65,535).
 - Specify the **Metric-Type**: **Type 1** or **Type 2** (default). Type E1 cost is the sum of the external cost plus the internal cost to reach that route. Type E2 is only the external

cost of that route. This can be useful when you want to load-balance the same external route, for example.

- Select a **Redistribute Route-Map** or [create a new Redistribute Route Map](#) whose match criteria control which connected routes to redistribute into OSPF. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route. Likewise, the Metric Type in the route map Set configuration takes precedence over the Metric Type configured in this redistribution profile.
5. Select **RIPv2** to allow configuration of this portion of the profile.
- **Enable** the RIPv2 portion of the profile.
 - Specify the **Metric** to apply to the RIPv2 routes being redistributed into OSPF (range is 0 to 4,294,967,295).
 - Specify the **Metric-Type: Type 1 or Type 2** (default).
 - Select a **Redistribute Route-Map** or [create a new Redistribute Route Map](#) whose match criteria control which RIPv2 routes to redistribute into OSPF. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route. Likewise, the Metric Type in the route map Set configuration takes precedence over the Metric Type configured in this redistribution profile.
6. Select **BGP AFI IPv4** to allow configuration of this portion of the profile.
- **Enable** the BGP AFI IPv4 portion of the profile.
 - Specify the **Metric** to apply to the BGP routes being redistributed into OSPF (range is 0 to 4,294,967,295).
 - Specify the **Metric-Type: Type 1 or Type 2** (default).
 - Select a **Redistribute Route-Map** or [create a new Redistribute Route Map](#) whose match criteria control which BGP IPv4 routes to redistribute into OSPF. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route. Likewise, the Metric Type in the route map Set configuration takes precedence over the Metric Type configured in this redistribution profile.
7. Select **IPv4 Default Route** to allow configuration of this portion of the profile.
- Select **Always** to always redistribute the IPv4 default route to OSPF; default is enabled.
 - **Enable** the IPv4 Default Route portion of the profile.
 - Specify the **Metric** to apply to the default route being redistributed into OSPF (range is 0 to 4,294,967,295).
 - Specify the **Metric-Type: Type 1 or Type 2** (default).
8. Click **OK**.

STEP 5 | Commit.

Configure OSPFv3 on an Advanced Routing Engine

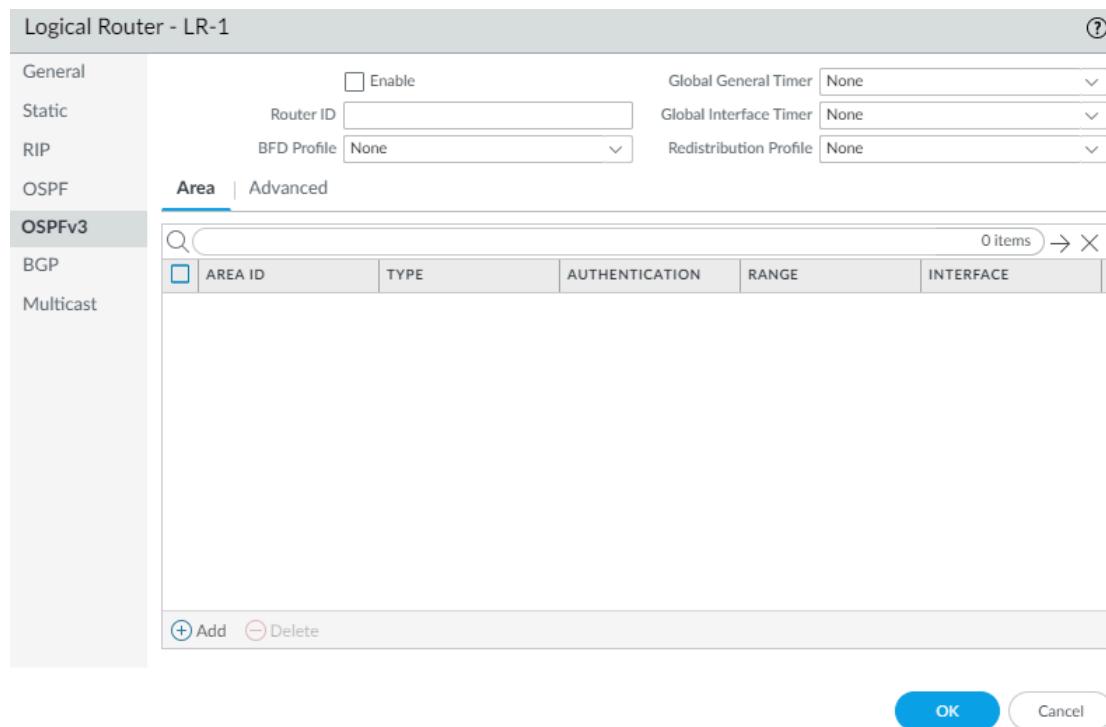
The Advanced Routing Engine supports OSPFv3, which supports only IPv6 addressing. Before you configure OSPFv3, you should understand [OSPF Concepts](#).

Consider the [OSPFv3 Routing Profiles](#) and [filters](#) that you can apply to OSPFv3 and thereby save configuration time and maintain consistency. You can create profiles and filters in advance or as you configure OSPFv3.

STEP 1 | Configure a Logical Router.

STEP 2 | Configure general OSPFv3 routing options.

1. Select **Network > Routing > Logical Routers** and select the logical router.
2. Select **OSPFv3** and **Enable** it.

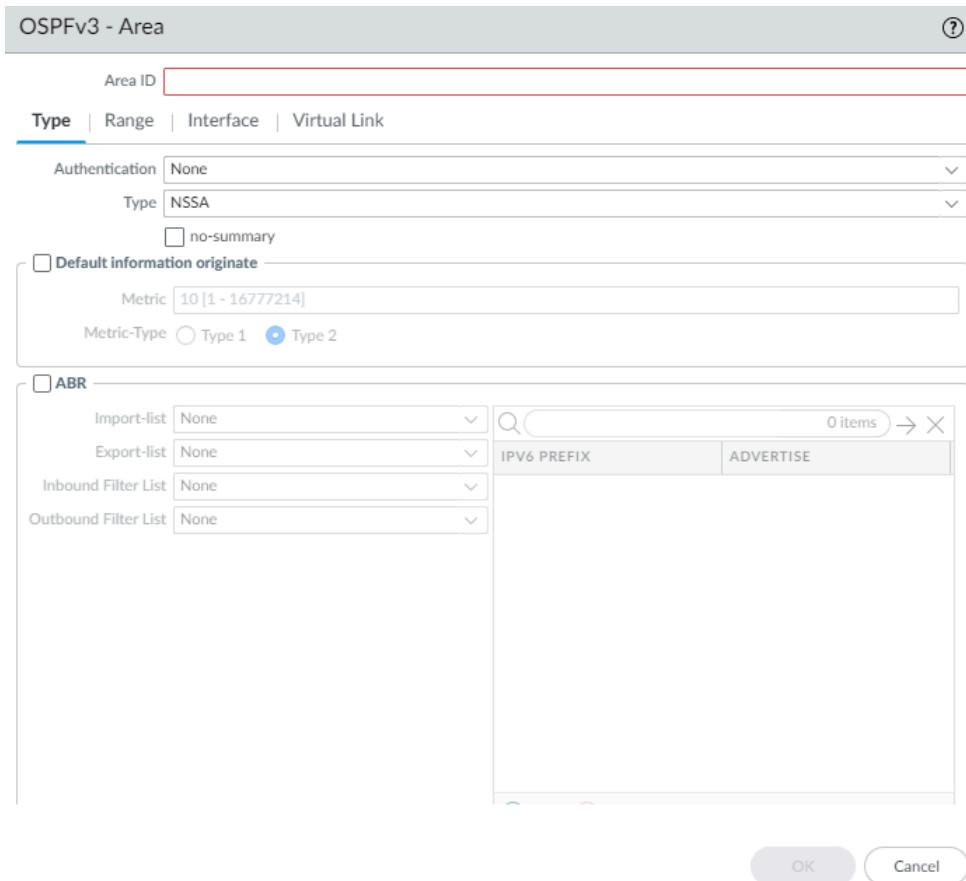


3. Assign a **Router ID** to OSPFv3 for the logical router, which is typically an IPv4 address (even though OSPFv3 is for IPv6 addressing), to ensure the Router ID is unique.
4. If you want to apply BFD to OSPFv3, select a **BFD Profile** you created, or select the **default** profile, or [create a new BFD Profile](#) to apply to all OSPFv3 interfaces belonging to the logical router. Default is **None (Disable BFD)**.
5. Select a **Global General Timer** profile or [create a new one](#) to set SPF throttle timers and to set the minimum interval between arriving instances of the same link-state advertisement (LSA).
6. Select a **Global Interface Timer** profile or [create a new one](#) to set the hello interval, retransmit interval, and other settings.
7. Select a **Redistribution Profile** or [create a new one](#) to redistribute IPv6 static routes, connected routes, IPv6 BGP routes, or the IPv6 default route to OSPFv3.
8. Click **OK**.

STEP 3 | Create an OSPFv3 area and specify characteristics based on the type of area.

1. Select **Network > Routing > Logical Routers** and select the logical router.
2. Select **OSPFv3 > Area** and **Add an Area by Area ID** (an IPv4 address).
3. On the **Type** tab, select an **Authentication** profile for the area or [create a new one](#).
4. Specify the **Type** of area:
 - **Normal**—There are no restrictions; the area can carry all types of routes.
 - **Stub**—There is no outlet from the area. To reach a destination outside of the area, traffic must go through an Area Border Router (ABR), which connects to other areas and area 0.
 - **NSSA (Not So Stubby Area)**—Traffic can leave the area directly, but only by using non-OSPF routes.
5. (**Stub and NSSA areas only**) Select **no-summary** to prevent the area from receiving Type 3 Summary LSAs and thereby reduce traffic in the area.
6. (**NSSA area only**) Select **Default information originate** to cause OSPFv3 to originate a default route.
 - Enter a **Metric** for the default route; range is 1 to 16,777,214; default is 10.
 - Select the **Metric-Type: Type 1 or Type 2**. Type E1 cost is the sum of the external cost plus the internal cost to reach that route. Type E2 is only the external cost of

that route. This can be useful when you want to load-balance the same external route, for example.



7. Select **ABR** if you want to configure filtering options.
8. Select an **Import-list** or [create a new Access List](#) to filter Type-3 LSAs; applies to paths announced into the specified area as Type-3 summary LSAs.
9. Select an **Export-list** or [create a new Access List](#) to filter Type-3 summary LSAs announced to other areas originated from intra-area paths from the specified area.
10. Select an **Inbound Filter List** or [create a new Prefix List](#) to filter Type-3 summary LSAs coming into the area.



If you apply an Import access list and Inbound prefix list, firewall uses an AND operation (both lists must be met).

11. Select an **Outbound Filter List** or [create a new Prefix List](#) to filter Type-3 summary LSAs from the area.

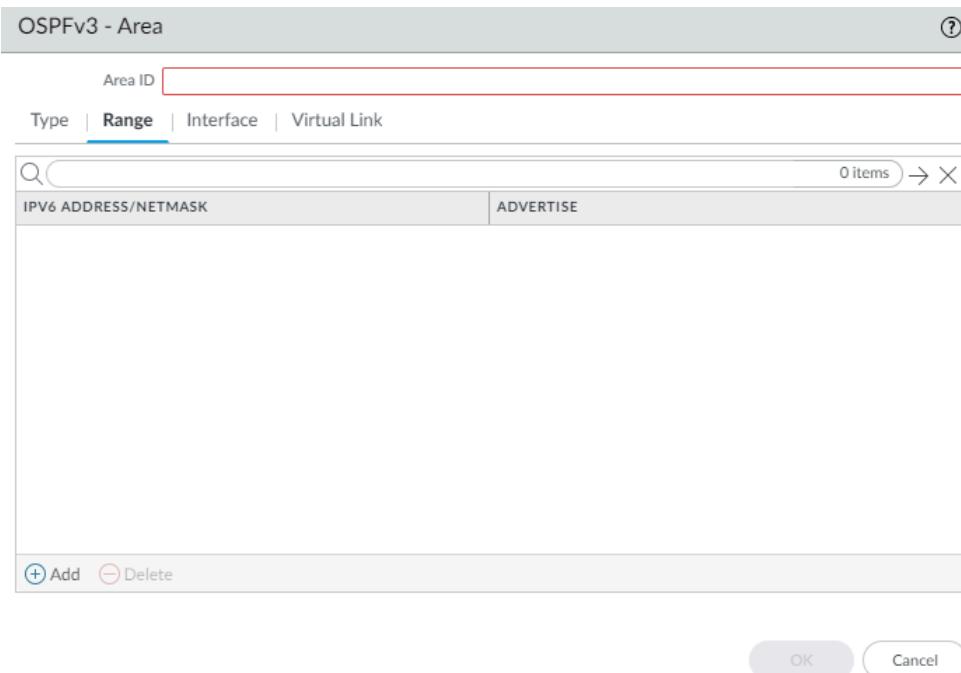


If you apply an Export access list and Outbound prefix list, firewall uses an AND operation (both lists must be met).

12. If the **Type** of area is **NSSA** and **ABR** is selected, **Add an IPv6 Prefix** to summarize a group of external subnets into a single Type-7 LSA, which is then translated to a Type-5 LSA and advertised to the backbone when you select **Advertise**.

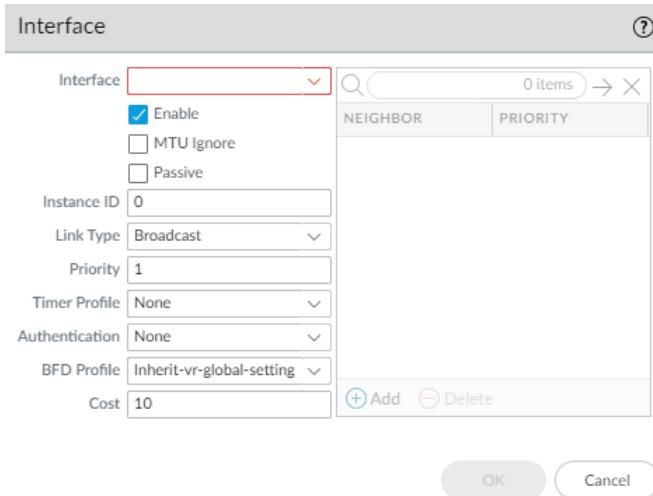
STEP 4 | Specify the network range that a Type-3 Summary LSA announces to the backbone area if the area contains at least one intra-area network (that is, described with router or network LSA) from this range.

1. Select **Range** and **Add an IPv6 Address/Netmask**, which summarizes routes for the area. A Type-3 Summary LSA with routing information that matches the range is announced into the backbone area if the area contains at least one intra-area network from this range.
2. Select **Advertise** to advertise matching subnets in LSAs to the backbone area. If **Advertise** is set to No, any matching intra-area prefixes that are present in the area will not be advertised in the backbone area.



STEP 5 | Add interfaces to the area.

1. On the **Interface** tab, **Add an Interface** by selecting one.
2. **Enable** the interface.



3. Select **MTU Ignore** to ignore maximum transmission unit (MTU) mismatches when trying to establish an adjacency (default is disabled; MTU match checking occurs).
4. Select **Passive** to prevent sending OSPF Hello packets out this interface and thus prevent the logical router from creating an OSPF adjacency with a neighbor; however, the interface is still included in the link-state database. You can make an interface passive, for example if it connects to a switch, because you don't want to send Hello packets where there is no router.
5. Keep the **Instance ID** set to 0 because only one instance of OSPFv3 is allowed.
6. Select the **Link Type**:
 - **Broadcast**—All neighbors that are accessible through the interface are discovered automatically by multicasting OSPF Hello messages, such as over an Ethernet interface.
 - **p2p** (point-to-point)—Automatically discover the neighbor.
 - **p2mp** (point-to-multipoint)—Neighbors must be defined manually: **Add the Neighbor IPv6 address** for all neighbors that are reachable through this interface and the

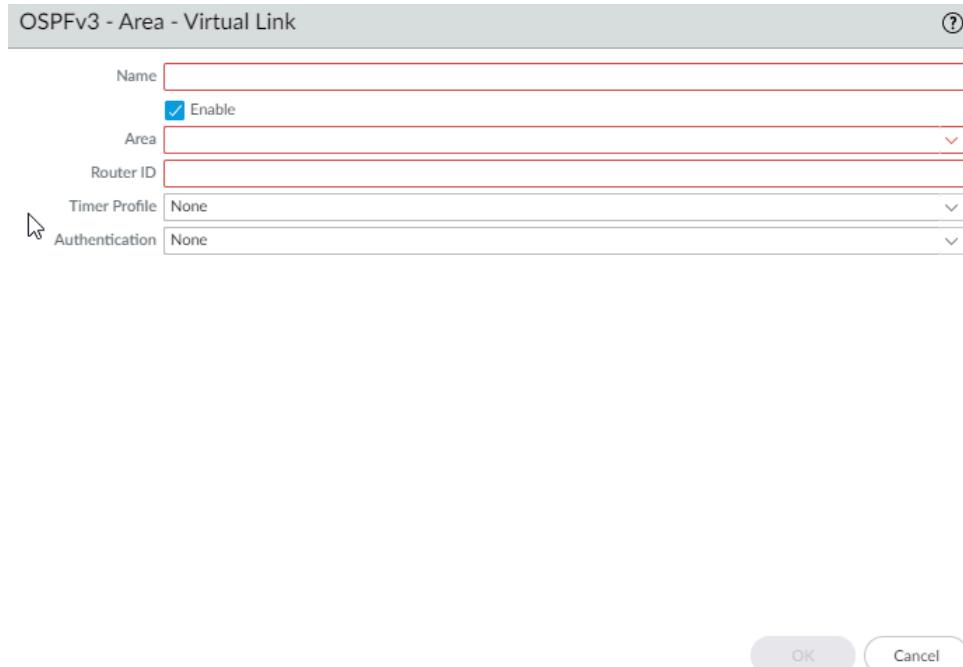
Priority of each neighbor to be elected the designated router (DR) or backup DR; range is 0 to 255; default is 1.

7. Enter a **Priority** for the interface—the priority for the router to be elected as a designated router (DR) or backup DR (BDR); range is 0 to 255; default is 1. If zero is configured, the router will not be elected as DR or BDR.
8. Select an OSPFv3 Interface **Timer Profile** or [create a new one](#) to apply to the interface. This OSPFv3 Interface Timer profile overrides the Global Interface Timer applied to OSPFv3.
9. Select an OSPFv3 Interface **Authentication** profile or [create a new one](#) to apply to the interface. This Authentication Profile overrides the Authentication Profile applied to the Area (on the Type tab).
10. By default, the interface will inherit the BFD profile you applied to the logical router for OSPFv3 (**Inherit-vr-global-setting**). Alternatively, select the **default** profile, select a **BFD Profile** you created, [create a new one](#), or select **None (Disable BFD)** to override the BFD Profile applied at the OSPFv3 level.
11. Enter an OSPFv3 **Cost** for the interface, which influences route selection; range is 1 to 65,5535; default is 10. During route selection, a route with a lower cumulative cost (the added costs of each interface used) is preferred over a route with a higher cumulative cost.
12. Click **OK** to save the interface.

STEP 6 | If the ABR does not have a physical link to the backbone area, configure a virtual link to a neighbor ABR within the same area that has a physical link to the backbone area.

 *The following settings must be defined for area border routers (ABRs) and must be defined within the backbone area (0.0.0.0).*

1. Select **Virtual Link**.
2. Add a **Virtual Link by Name** (a maximum of 31 characters).
3. **Enable** the virtual link.



The screenshot shows the 'OSPFv3 - Area - Virtual Link' configuration dialog. It includes fields for Name, Enable (checked), Area (selected), Router ID, Timer Profile (None), and Authentication (None). At the bottom are OK and Cancel buttons.

4. Select the transit **Area** where the neighbor ABR that has the physical link to the backbone area is located.
5. Enter the **Router ID** of the neighbor ABR on the remote end of the virtual link.
6. Select an OSPFv3 Interface **Timer Profile** or [create a new Timer Profile](#) to apply to the virtual link. This OSPFv3 Interface Timer profile overrides the Global Interface Timer applied to OSPFv3 and the OSPFv3 Interface Timer profile applied to the interface.
7. Select an OSPF Interface **Authentication** profile or [create a new Authentication Profile](#) to apply to the virtual link. This Authentication Profile overrides the Authentication Profile applied to the Area (on the Type tab) and the Authentication Profile applied to the interface.
8. Click **OK**.

STEP 7 | Click **OK** to save the area.

STEP 8 | Configure advanced OSPFv3 features.

1. Select **Network > Routing > Logical Routers** and select the logical router.
2. Select **OSPFv3 > Advanced**.
3. **Enable Graceful Restart** to enable Graceful Restart for the logical router. Default is enabled.
4. **Enable Helper Mode** to enable the logical router to function in Graceful Restart helper mode. Default is enabled.
5. **Enable Strict LSA Checking** to cause the helper router to stop performing helper mode and to cause the graceful restart process to stop if a link-state advertisement indicates a network topology change. Default is enabled.
6. Enter a **Grace Period (sec)**—the number of seconds within which the logical router will perform a graceful restart if the firewall goes down or becomes unavailable; range is 5 to 1,800; default is 120.
7. Enter a **Max Neighbor Restart Time (sec)**—the maximum number of seconds of Grace Period that the logical router accepts from a neighbor when the logical router is in Helper Mode; range is 5 to 1,800; default is 140.
8. Select **Disable R-Bit and v6-Bit** to clear the R-bit and V6-bit in router LSAs sent from this logical router to indicate that the firewall is not active. When in this state, the firewall participates in OSPFv3 but does not send transit traffic or IPv6 datagrams. In this state, local traffic will still be forwarded to the firewall. This is useful while performing maintenance with a dual-homed network because traffic can be re-routed around the firewall while it can still be reached. See [RFC 5340](#).

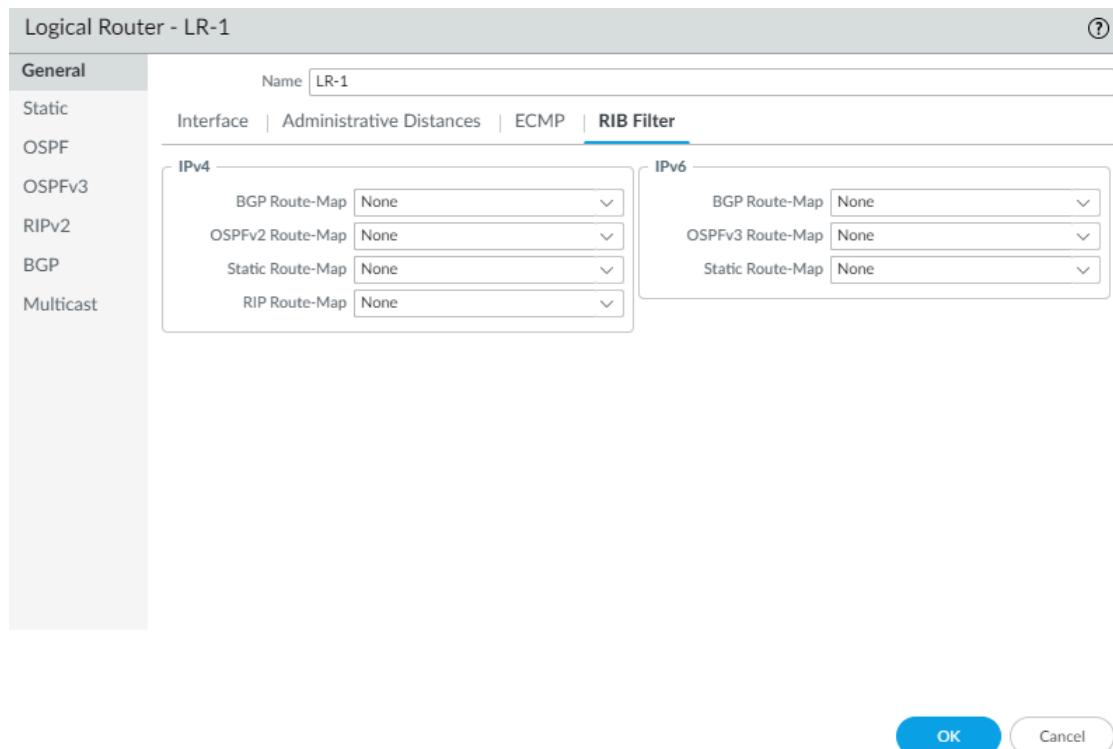
The screenshot shows the 'Logical Router - LR-1' configuration dialog. The 'Advanced' tab for OSPFv3 is selected. In the 'Graceful Restart' section, three checkboxes are checked: 'Enable Graceful Restart', 'Enable Helper Mode', and 'Enable Strict LSA Checking'. Below these are two input fields: 'Grace Period (sec)' set to 120 and 'Max Neighbor Restart Time (sec)' set to 140. A checkbox for 'Disable R-Bit and v6-Bit' is present but unchecked. At the bottom right are 'OK' and 'Cancel' buttons.

9. Click **OK** to save advanced settings.

STEP 9 | Configure intra-area filtering to determine which OSPFv3 routes are placed in the global RIB.

You might learn OSPFv3 routes and redistribute them, but not want them in the global RIB; you might want to allow only specific OSPFv3 routes to the global RIB.

1. Select **Network > Routing > Logical Routers** and select a logical router.
2. Select **RIB Filter**.
3. To filter IPv6 OSPFv3 routes for the global RIB, for **OSPFv3 Route-Map**, select a Redistribution route map you created or [create a new Redistribution Route Map](#) in which the Source Protocol is OSPFv3 and the Destination Protocol is RIB.



4. Click **OK**.

STEP 10 | (Optional) Change the default administrative distances for OSPFv3 Intra Area, OSPFv3 Inter Area, and OSPFv3 External Routes that pertain to the [logical router](#).

STEP 11 | Commit.

STEP 12 | View advanced routing information for OSPFv3 and the link-state database (LSDB). The PAN-OS CLI Quick Start lists the commands in the [CLI Cheat Sheet: Networking](#).

Create OSPFv3 Routing Profiles

The Advanced Routing Engine supports OSPFv3; create OSPFv3 global timer profiles, authentication profiles, interface timer profiles, and redistribution profiles to apply to OSPFv3. This topic describes the profiles and how to create them. Reference them when you [Configure OSPFv3 on an Advanced Routing Engine](#).

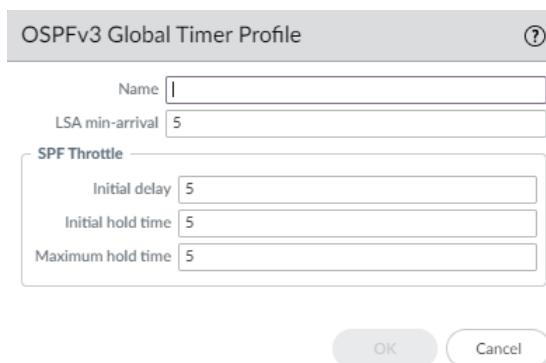
- **OSPFv3 Global Timer Profiles**—Specify the timers for the link-state advertisement (LSA) interval, SPF calculation delay, initial hold time, and maximum hold time that apply all OSPFv3 areas. SPF Throttle settings allow the protocol to slow the sending of LSA updates while a network is unstable (undergoing topology changes). Apply the profile in the general OSPFv3 configuration. The profile is global for OSPFv3 on the logical router; you can create more than one to easily change global timers.
- **OSPFv3 Interface Authentication Profiles**—OSPFv3 does not have its own authentication capabilities; it relies on IPSec to secure OSPFv3 messages between neighbors. Apply the profile in the **OSPFv3 Area > Type** tab.
- **OSPFv3 Interface Timer Profiles**—Specify timers related to interface operations, such as OSPFv3 hello and graceful restart. Apply the profile in the general OSPFv3 configuration.
- **OSPFv3 Redistribution Profiles**—Redistribute IPv6 static, connected, or IPv6 BGP routes or the IPv6 default route into OSPFv3. Apply the profile in the general OSPFv3 configuration.

STEP 1 | Create an OSPFv3 Global Timer Profile.

1. Select **Network > Routing > Routing Profiles > OSPFv3**.
2. **Add an OSPFv3 Global Timer Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain

a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.

3. Enter the **LSA min-arrival** (in seconds), which is the smallest interval at which the firewall recalculates the SPF tree; range is 1 to 10; default is 5. The firewall would recalculate at a larger interval (less frequently than the setting).
4. In the SPF Throttle area, enter the **Initial delay** (in seconds) from when the logical router receives a topology change until it performs the Shortest Path First (SPF) calculation; range is 0 to 600; default is 5.
5. Enter the **Initial hold time** (in seconds) between the first two consecutive SPF calculations; range is 0 to 600; default is 5. Each subsequent hold time is twice as long as the prior hold time until the hold time reaches the maximum hold time.
6. Enter the **Maximum hold time** (in seconds), which is the largest value that the hold time increases to until it remains steady; range is 0 to 600; default is 5.



7. Click **OK**.

STEP 2 | Create an OSPFv3 Interface Authentication Profile.

1. Select **Network > Routing > Routing Profiles > OSPFv3**.
2. **Add an OSPFv3 Auth Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a

combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.

3. Enter the **SPI** (Security Policy Index), which must match between both ends of the OSPFv3 adjacency.
4. Select the **Protocol: ESP** (Encapsulating Security Payload) (recommended) or **AH** (Authentication Header).
5. Select the **Type** of authentication:
 - **SHA1** (default) Secure Hash Algorithm 1
 - **SHA256**
 - **SHA384**
 - **SHA512**
 - **MD5**
 - **None**
6. Enter the authentication **Key** using 5 hexadecimal sections of 8 hexadecimal characters for a total of 40 hexadecimal characters (for example, A5DEC4DD155A695A8B983AACCEAA5A97C6AECB6D1).
7. **Confirm Key** by entering the same key.

The dialog box is titled "OSPFv3 Auth Profile". It contains fields for "Name" and "SPI", and a "Protocol" section with radio buttons for "ESP" (selected) and "AH". Below this is an "Authentication" section with dropdown menus for "Type" (set to "SHA1") and "Key", and a "Confirm Key" field. At the bottom are "Encryption" settings for "Algorithm" (set to "3des"), "Key", and "Confirm Key". At the very bottom are "OK" and "Cancel" buttons.

8. **(ESP only)** Select the encryption **Algorithm**:
 - **3des** (default)
 - **aes-128-cbc**
 - **aes-192-cbc**
 - **aes-256-cbc**
 - **null**
9. Enter the encryption **Key** in hexadecimal format; use the correct number of sections based on the type of ESP encryption:
 - **3des**—Use a total of 6 hexadecimal sections in the key.
 - **aes-128-cbc**—Use a total of 4 hexadecimal sections in the key.
 - **aes-192-cbc**—Use a total of 6 hexadecimal sections in the key.
 - **aes-256-cbc**—Use a total of 8 hexadecimal sections in the key.
10. **Confirm Key** by entering the same key.

11. Click **OK**.

STEP 3 | Create an OSPFv3 Interface Timer Profile.

1. Select **Network > Routing > Routing Profiles > OSPFv3**.
2. **Add an OSPFv3 Interface Timer Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.

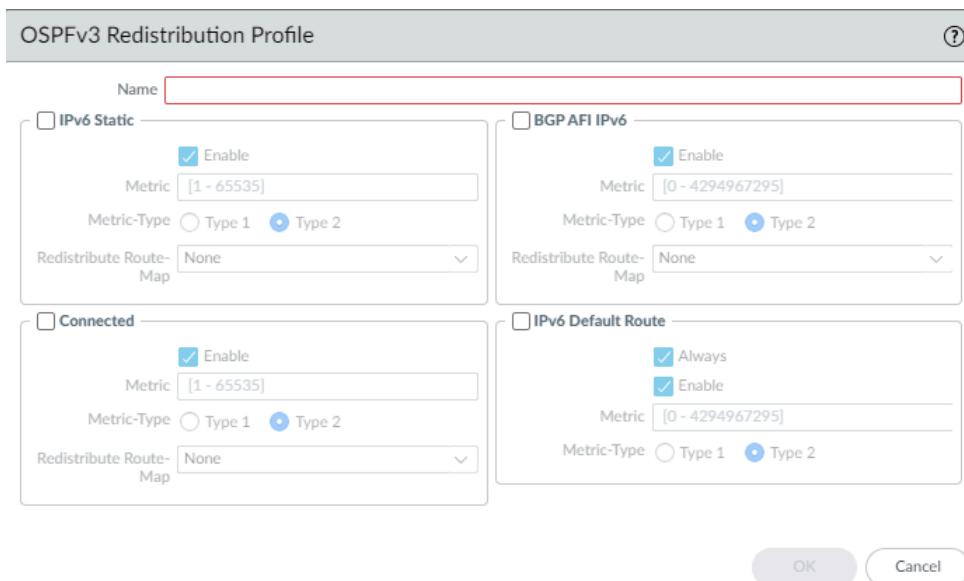
OSPFv3 Interface Timer Profile	
Name	<input type="text"/>
Hello Interval	10
Dead Count	4
Retransmit Interval	5
Transmit Delay	1
Graceful Restart Hello Delay (sec)	10

OK **Cancel**

3. Enter the **Hello Interval**, the interval (in seconds) at which OSPFv3 sends Hello packets; range is 1 to 3,600; default is 10.
4. Enter the **Dead Count**, the number of times the Hello Interval can occur from a neighbor without OSPFv3 receiving a Hello packet from the neighbor, before OSPFv3 considers that neighbor down; range is 3 to 20; default is 4.
5. Enter the **Retransmit Interval**, the number of seconds that OSPFv3 waits to receive an ACK for an LSA from a neighbor before OSPFv3 retransmits the LSA; range is 1 to 1,800; default is 5.
6. Enter the **Transmit Delay**, the number of seconds that OSPFv3 delays transmitting an LSA before sending the LSA out an interface; range is 1 to 1,800; default is 1.
7. Enter the **Graceful Restart Hello Delay (sec)** in seconds; range is 1 to 10; default is 10. This setting applies to an OSPFv3 interface when Active/Passive HA is configured. Graceful Restart Hello Delay is the number of seconds during which the firewall sends Grace LSA packets at 1-second intervals. During this time, no Hello packets are sent from the restarting firewall. During the restart, the dead time (which is the **Hello Interval** multiplied by the **Dead Count**) is also counting down. If the dead timer is too short, the adjacency will go down during the graceful restart because of the hello delay. Therefore it is recommended that the dead timer be at least four times the value of the Graceful Restart Hello Delay. For example, a **Hello Interval** of 10 seconds and a **Dead Count** of 4 yield a dead timer of 40 seconds. If the **Graceful Restart Hello Delay** is set to 10 seconds, that 10-second delay of hello packets is comfortably within the 40-second dead timer, so the adjacency will not time out during a graceful restart.
8. Click **OK**.

STEP 4 | Create an OSPFv3 Redistribution Profile to specify any combination of IPv6 static routes, connected routes, IPv6 BGP routes, and default IPv6 route to redistribute to OSPFv3.

1. Select Network > Routing > Routing Profiles > OSPFv3.
2. Add an **OSPFv3 Redistribution Profile by Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.



3. Select **IPv6 Static** to allow configuration of this portion of the profile.
 - **Enable** the IPv6 static redistribution portion of the profile.
 - Enter a **Metric** to apply to the IPv6 static routes redistributed to OSPFv3; range is 1 to 65,535.
 - Select a **Metric Type**: **Type 1** or **Type 2**.
 - Select a **Redistribute Route-Map** or [create a new Redistribution Route Map](#) whose Match criteria control the IPv6 static routes to redistribute into OSPFv3. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route. Likewise, the Metric Type in the route map Set configuration takes precedence over the Metric Type configured in this redistribution profile.
4. Select **Connected** to allow configuration of this portion of the profile.
 - **Enable** the connected route redistribution portion of the profile.
 - Enter a **Metric** to apply to the connected routes redistributed to OSPFv3; range is 1 to 65,535.
 - Select a **Metric Type**: **Type 1** or **Type 2**.
 - Select a **Redistribute Route-Map** or [create a new Redistribution Route Map](#) whose Match criteria control the connected routes to redistribute into OSPFv3. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this

redistribution profile is applied to the redistributed route. Likewise, the Metric Type in the route map Set configuration takes precedence over the Metric Type configured in this redistribution profile.

5. Select **BGP AFI IPv6** to allow configuration of this portion of the profile.
 - **Enable** the BGP AFI IPv6 route redistribution portion of the profile.
 - Enter a **Metric** to apply to the IPv6 BGP routes redistributed to OSPFv3; range is 0 to 4,294,967,295.
 - Select a **Metric Type: Type 1 or Type 2**.
 - Select a **Redistribute Route-Map** or [create a new Redistribution Route Map](#) whose Match criteria control the IPv6 BGP routes to redistribute into OSPFv3. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route. Likewise, the Metric Type in the route map Set configuration takes precedence over the Metric Type configured in this redistribution profile.
6. Select **IPv6 Default Route** to allow configuration of this portion of the profile.
 - Select **Always** to always create and redistribute the default route to OSPFv3, even if there is no default route on the router; default is enabled. If **Always** is not set, when there is no default route on the ABR, the default route is not redistributed.
 - **Enable** the IPv6 Default Route redistribution portion the profile.
 - Enter a **Metric** to apply to the IPv6 default route redistributed to OSPFv3; range is 0 to 4,294,967,295.
 - Select a **Metric Type: Type 1 or Type 2**.
7. Click **OK**.

STEP 5 | Commit.

Configure RIPv2 on an Advanced Routing Engine

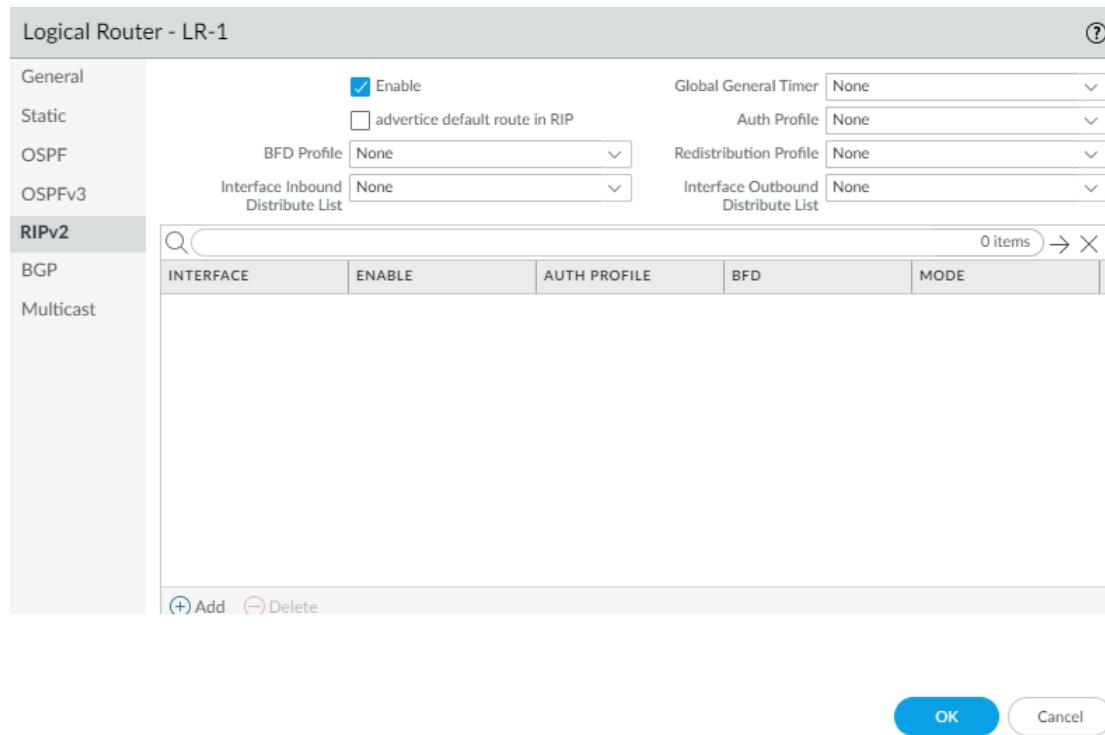
The Advanced Routing Engine supports RIPv2.

Consider the [RIPv2 Routing Profiles](#) and [filters](#) that you can apply to RIPv2 and thereby save configuration time and maintain consistency. You can create profiles and filters in advance or as you configure RIPv2.

STEP 1 | Configure a Logical Router.

STEP 2 | Enable RIPv2 and configure general settings.

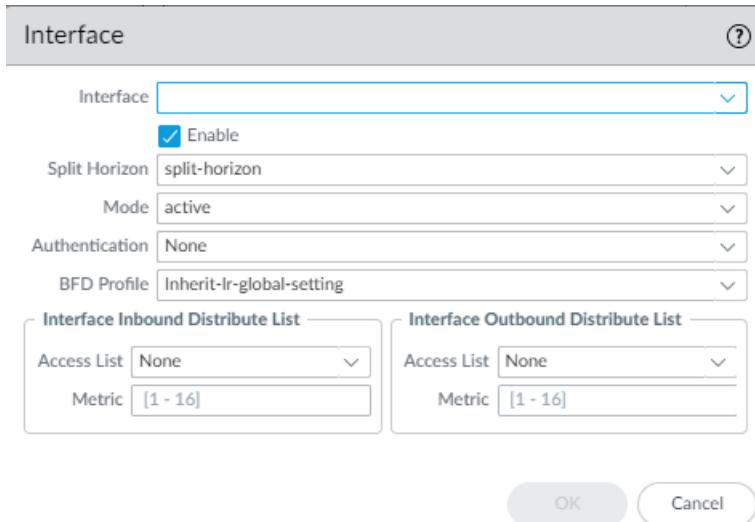
1. Select **Network > Routing > Logical Routers** and select a logical router.
2. Select **RIPv2** and **Enable** it.



3. Select **advertise default route in RIP** to advertise the default route even if it doesn't exist in the RIB of the routing engine.
4. If you want to apply BFD to RIPv2, select a **BFD Profile** you created, or select the **default** profile, or [create a new BFD Profile](#). Default is **None (Disable BFD)**.
5. Select a **Global General Timer** or [create a new RIPv2 Global Timer Profile](#).
6. Select an **Auth Profile** or [create a new RIPv2 Authentication Profile](#).
7. Select a **Redistribution Profile** or [create a new Redistribution Profile](#) to redistribute IPv4 static routes, connected routes, BGP IPv4 routes, or OSPFv2 routes to RIPv2.
8. Select a **Global Inbound Distribute List** to control the incoming routes accepted.
9. Select an **Global Outbound Distribute List** to control the routes advertised to RIP neighbors.

STEP 3 | Configure an interface for RIPv2.

1. Add an Interface by selecting one and Enable it.



2. For **Split Horizon**, select one of the following:

- **split-horizon**—Does not advertise a route back on the same interface where it was received.
- **no-split-horizon**—Disables split horizon.
- **no-split-horizon-with-poison-reverse**—Allows the advertisement back on the same interface where it was received and sets the metric for these routes to the maximum allowed for RIP, which is 16.

3. Select the **Mode**:

- **active**—The interface will advertise networks and send RIP updates.
- **passive**—The interface will advertise networks, but not send RIP updates. (Useful if there are no RIP routers for the network, and therefore no reason to send RIP updates on the interface.)
- **send-only**—Can be used if the firewall is an end node and you only want to advertise a prefix to RIP, but use static routes or a default route to reach external prefixes.

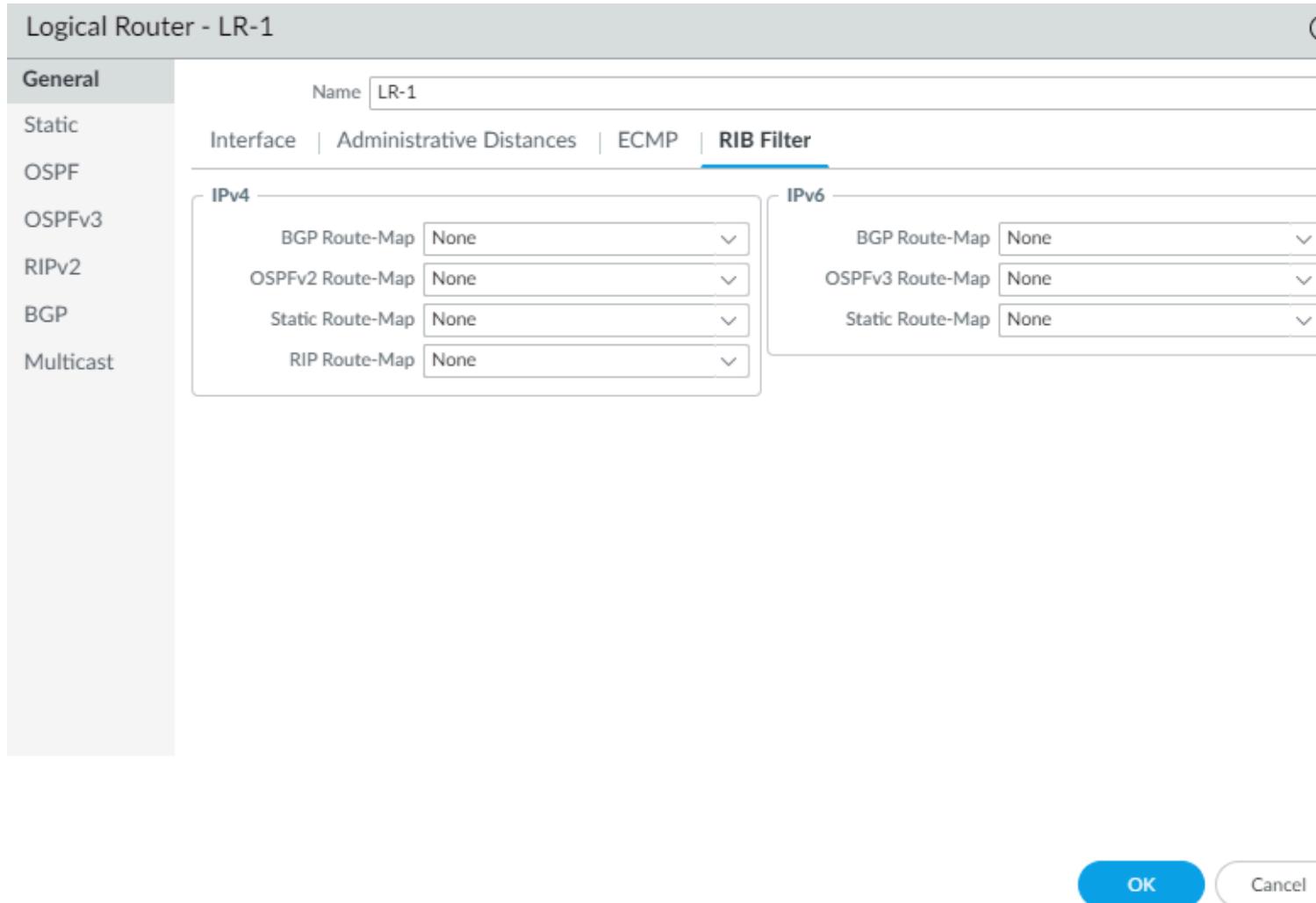
4. Select an **Authentication** profile if you want to override the profile you applied at the logical router level.
5. By default, the interface will inherit the BFD profile you applied to the logical router for RIPv2 (**Inherit-Ir-global-setting**). Alternatively, select a different **BFD Profile**, [create a new BFD Profile](#), or select **None (Disable BFD)** to disable BFD for the interface.
6. For **Interface Inbound Distribute List**, select an **Access List** to control the routes coming to this interface.
7. Specify the **Metric** applied to incoming routes; range is 1 to 16.
8. For **Interface Outbound Distribute List**, select an **Access List** to control the routes advertised out this interface to RIP neighbors.
9. Specify the **Metric** to apply to advertised routes; range is 1 to 16.
10. Click **OK**.

STEP 4 | Click **OK**.

STEP 5 | (Optional) Control RIP routes that are placed in the global RIB.

You might learn routes and redistribute them, but not want them in the protocol's local route table or global RIB. You might want to add only specific routes to the global RIB.

1. Select **Network > Routing > Logical Routers** and select a logical router.
2. Select **RIB Filter** to allow routes into or prevent routes from being added to the global RIB.



3. To filter RIPv2 routes going to the RIB, in the IPv4 area, for **RIP Route-Map**, select a Redistribution Route Map or [create a new one](#).
4. Click **OK**.

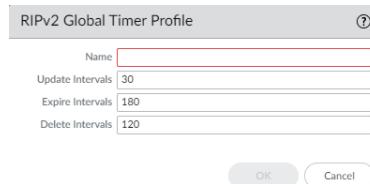
Create RIPv2 Routing Profiles

The Advanced Routing Engine supports RIPv2; create the following profiles to apply to the protocol. The profiles can be used across multiple logical routers and virtual systems. This topic describes the profiles and how to configure them.

- **RIPv2 Global Timer Profiles**—Specify RIPv2 update, expire, and delete intervals. Apply the profile in the RIPv2 general configuration.
- **RIPv2 Interface Authentication Profiles**—Specify RIPv2 authentication using a password or MD5; apply the profile in the RIPv2 general configuration.
- **RIPv2 Redistribution Profiles**—Specify how to redistribute IPv4 static routes, connected routes, BGP IPv4 routes, and OSPFv2 routes to RIPv2. Apply the profile in the RIPv2 general configuration.

STEP 1 | Create a RIPv2 Global Timer Profile.

1. Select **Network > Routing > Routing Profiles > RIPv2**.
2. Add a **RIPv2 Global Timer Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.

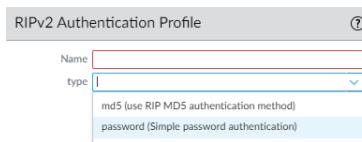


3. Specify **Update Interval** in seconds, which is the length of time between regularly scheduled Update messages; range is 5 to 2,147,483,647; default is 30.
4. Specify **Expire Interval** in seconds, which is the length of time that a route can be in the routing table without being updated; range is 5 to 2,147,483,647; default is 180. After the Expire Interval is reached, the route is still included in Update messages until the Delete Interval is reached.
5. Specify **Delete Interval** in seconds; range is 5 to 2,147,483,647; default is 120. When an expired route in the routing table reaches the Delete Interval, it is deleted from the routing table.
6. Click **OK**.

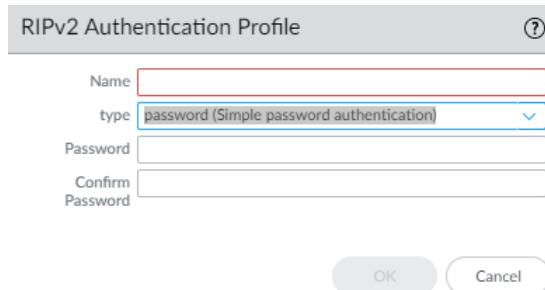
STEP 2 | Create a RIPv2 Authentication Profile.

1. Select **Network > Routing > Routing Profiles > RIPv2**.
2. Add a **RIPv2 Authentication Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain

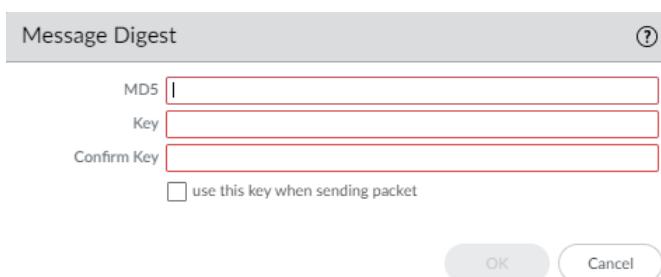
a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.



3. Specify the **type** of authentication: **md5 (use RIP MD5 authentication method)** or **password (Simple password authentication)**.
4. For **Simple password authentication**, enter the **Password** (a maximum of 16 characters) and **Confirm Password**.



5. For **RIP MD5 authentication**:
 - Add an MD5 Key-ID; range is 0 to 255.
 - Enter the **Key** (a maximum of 16 alphanumeric characters) and **Confirm Key**.
 - Select **use this key when sending packet** to make this key the Preferred key.

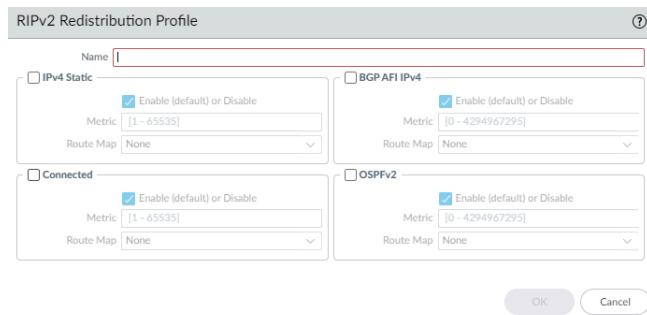


6. Click **OK**.

STEP 3 | Create a RIPv2 Redistribution Profile to specify any combination of IPv4 static routes, connected routes, BGP IPv4 routes, and OSPFv2 routes to redistribute to RIPv2.

1. Select **Network > Routing > Routing Profiles > RIPv2**.
2. **Add a RIPv2 Redistribution Profile by Name** (a maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain

a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.



3. Select **IPv4 Static** to allow configuration of this portion of the profile.
 - **Enable** the IPv4 static redistribution portion of the profile.
 - Specify the **Metric** to apply to the static routes being redistributed into RIPv2 (range is 1 to 65,535).
 - Select a **Redistribute Route-Map** or [create a new Redistribute Route Map](#) whose match criteria control which IPv4 static routes to redistribute into RIPv2. Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route.
4. Select **Connected** to allow configuration of this portion of the profile.
 - **Enable** the connected route redistribution portion of the profile.
 - Specify the **Metric** to apply to the connected routes being redistributed into RIPv2 (range is 1 to 65,535).
 - Select a **Redistribute Route-Map** or [create a new Redistribute Route Map](#). Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route.
5. Select **BGP AFI IPv4** to allow configuration of this portion of the profile.
 - **Enable** the BGP IPv4 route redistribution portion of the profile.
 - Specify the **Metric** to apply to the BGP routes being redistributed into RIPv2 (range is 0 to 4,294,967,295).
 - Select a **Redistribute Route-Map** or [create a new Redistribute Route Map](#). Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value, they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route.
6. Select **OSPFv2** to allow configuration of this portion of the profile.
 - **Enable** the OSPFv2 route redistribution portion of the profile.
 - **Enable** the IPv4 default route redistribution portion of the profile.
 - Specify the **Metric** to apply to the default route being redistributed into RIPv2 (range is 0 to 4,294,967,295).
 - Select a **Redistribute Route-Map** or [create a new Redistribute Route Map](#). Default is **None**. If the route map Set configuration includes a Metric Action and Metric Value,

they are applied to the redistributed route. Otherwise, the Metric configured on this redistribution profile is applied to the redistributed route.

7. Click **OK**.

Create BFD Profiles

On an Advanced Routing Engine, you can use Bidirectional Forwarding Detection (BFD) profiles to easily apply BFD settings to a static route or routing protocol. You can use the default profile (which is read-only) or create new BFD profiles.

Perform the following before creating a BFD profile:

- [Configure a Logical Router](#).
- Configure one or more static routes if you are applying BFD to a static route.
- Configure a routing protocol (**BGP**, **OSPF**, **OSPFv3**, or **RIPv2**) if you are applying BFD to a routing protocol. For example, you can apply a BFD profile when configuring general BGP settings.



The effectiveness of your BFD implementation depends on various factors, such as traffic loads, network conditions, how aggressive your BFD settings are, and how busy the dataplane is.

STEP 1 | Select **Network > Routing > Routing Profiles > BFD**.

STEP 2 | Add a BFD profile by **Name** (maximum of 63 characters). The name is case-sensitive and must be unique on the firewall. Use only letters, numbers, hyphens, and underscores. No dot (.) or space is allowed.

BFD Profile	
Name	<input type="text"/>
Mode	<input checked="" type="radio"/> Active <input type="radio"/> Passive
Desired Minimum Tx Interval (ms)	1000
Desired Minimum Rx Interval (ms)	1000
Detection Time Multiplier	3
Hold Time (ms)	0
<input type="checkbox"/> Enable Multihop	
Minimum Rx TTL [1 - 254]	
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

STEP 3 | Select the **Mode** in which BFD operates:

- **Active**—BFD initiates sending control packets to peer (default). At least one of the BFD peers must be Active; both can be Active.
- **Passive**—BFD waits for peer to send control packets and responds as required.

STEP 4 | Enter the **Desired Minimum Tx Interval (ms)**, the minimum interval, in milliseconds, at which you want the BFD protocol to send BFD control packets; you are thus negotiating the transmit interval with the peer. Range for PA-7000 Series, PA-5200 Series, and PA-5450

firewall is 50 to 10,000; range for PA-3200 Series is 100 to 10,000; range for VM-Series is 200 to 10,000. Default is 1,000.

- If you have multiple routing protocols that use different BFD profiles on the same interface, configure the BFD profiles with the same **Desired Minimum Tx Interval**.
- On a PA-7000 Series firewall, set the **Desired Minimum Tx Interval** to 100 or greater; a value less than 100 is at risk of causing BFD flaps.

STEP 5 | Enter the **Required Minimum Rx Interval (ms)**. This is the minimum interval, in milliseconds, at which BFD can receive BFD control packets. Range for PA-7000 Series, PA-5200 Series, and PA-5450 firewall is 50 to 10,000; range for PA-3200 Series is 100 to 10,000; range for VM-Series is 200 to 10,000. Default is 1,000.

- On a PA-7000 Series firewall, set the **Desired Minimum Rx Interval** to 100 or greater; a value less than 100 is at risk of causing BFD flaps.

STEP 6 | Enter the **Detection Time Multiplier**. Range is 2 to 255, default is 3.

The local system calculates the detection time as the **Detection Time Multiplier** received from the remote system multiplied by the agreed transmit interval of the remote system (the greater of the **Required Minimum Rx Interval** and the last received **Desired Minimum Tx Interval**). If BFD does not receive a BFD control packet from its peer before the detection time expires, a failure has occurred.

- When creating a BFD profile, take into consideration that the firewall is a session-based device typically at the edge of a network or data center and may have slower links than a dedicated router. Therefore, the firewall likely needs a longer interval and a higher multiplier than the fastest settings allowed. A detection time that is too short can cause false failure detections when the issue is really just traffic congestion.

STEP 7 | Enter the **Hold Time (ms)**, the delay, in milliseconds, after a link comes up before BFD transmits BFD control packets. **Hold Time** applies to BFD **Active** mode only. If BFD receives BFD control packets during the Hold Time, it ignores them. Range is 0 to 120,000; default is 0, which means no transmit **Hold Time** is used; BFD sends and receives BFD control packets immediately after the link is established.

STEP 8 | Enter the **Minimum Rx TTL**, the minimum Time-to-Live (number of hops) BFD will accept (receive) in a BFD control packet when BGP supports multihop BFD. Range is 1 to 254; there is no default.

The firewall drops the packet if it receives a smaller TTL than its configured **Minimum Rx TTL**. For example, if the peer is 5 hops away and the peer transmits a BFD packet with a TTL of 100 to the firewall, and if the **Minimum Rx TTL** for the firewall is set to 96 or higher, the firewall drops the packet.

STEP 9 | Click **OK**.

Configure IPv4 Multicast

The Advanced Routing Engine supports IPv4 multicast for a logical router. You should be familiar with [IP Multicast](#), [IGMP](#), and [PIM](#) concepts.

IPv4 multicast on an Advanced Routing Engine supports features not supported on the legacy routing engine:

- IGMP static join, which is the ability to specify a static IGMPv3 or IGMPv2 receiver on an interface. The corresponding PIM Join message is sent upstream.
- Protocol Independent Multicast (PIM) supports Reverse-Path Forwarding (RPF) lookup modes: MRIB only, URIB only, and MRIB-then-URIB.

IPv4 multicast does not support IGMPv1.

When you configure IPv4 multicast, you [Create Multicast Routing Profiles](#) for PIM interface timers and IGMP interface queries to make your configuration easier and consistent. You can [create multicast route maps](#) to control PIM group permissions.

You can also [Create an IPv4 MRoute](#) if you want unicast traffic to take a different route from multicast traffic.

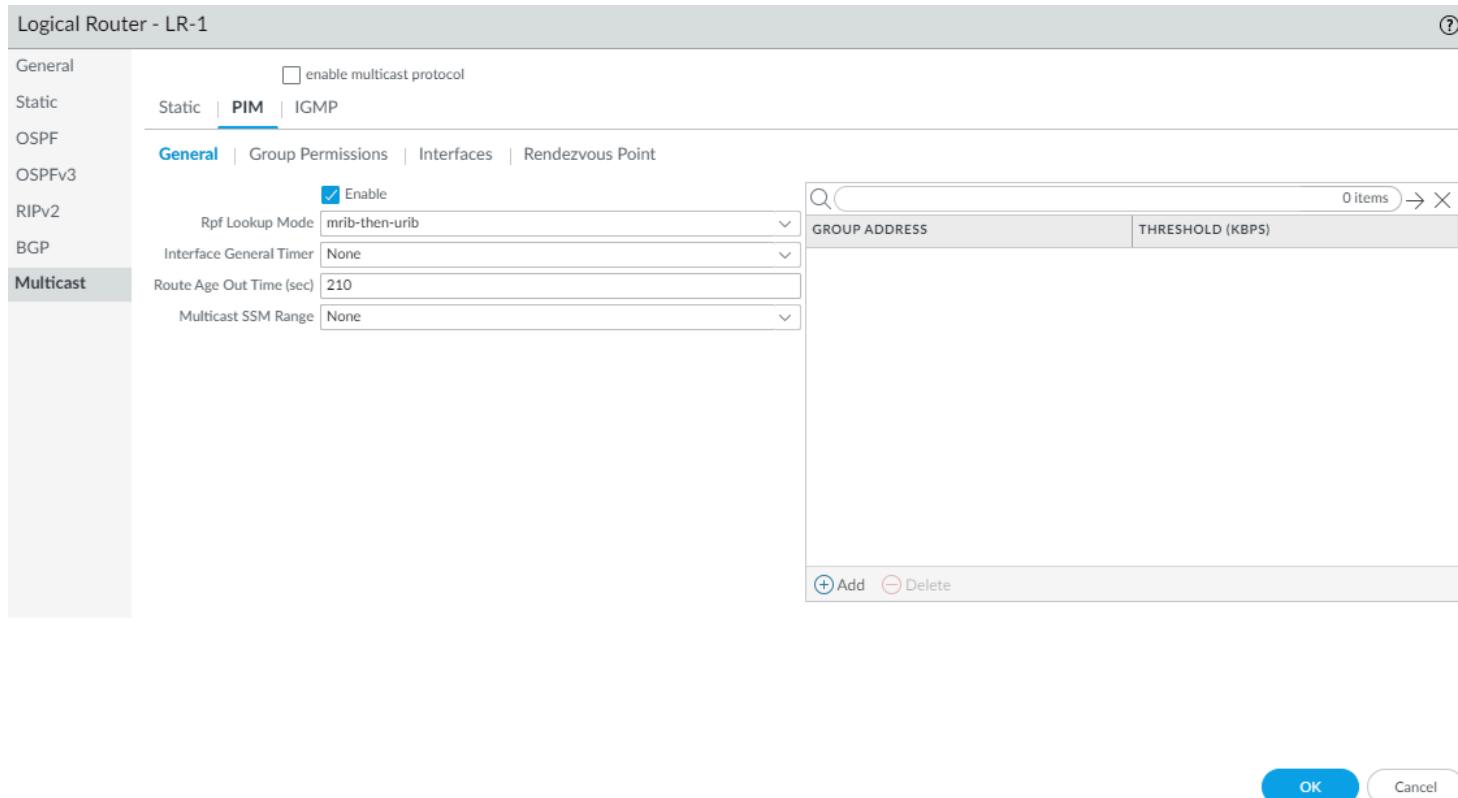
STEP 1 | Configure a Logical Router.

STEP 2 | Select **Network > Routing > Logical Routers** and select a logical router.

STEP 3 | Select **Multicast** and **enable multicast protocol**.

STEP 4 | Configure general PIM parameters for the logical router.

1. Select **PIM > General** and **Enable PIM**.



2. Select the **RPF lookup mode**, which determines where the logical router looks to find the outgoing interface to reach the source address contained in the multicast packet. If the outgoing interface stored in the RIB matches the interface on which the multicast

packet arrived, the logical router accepts and forwards the packet; otherwise, it drops the packet.

- **mrib-only**—Look in multicast RIB only.
- **mrib-then-urib**—Look in multicast RIB first; if route is not present in multicast RIB, then look in unicast RIB.
- **urib-only**—Look in unicast RIB only.

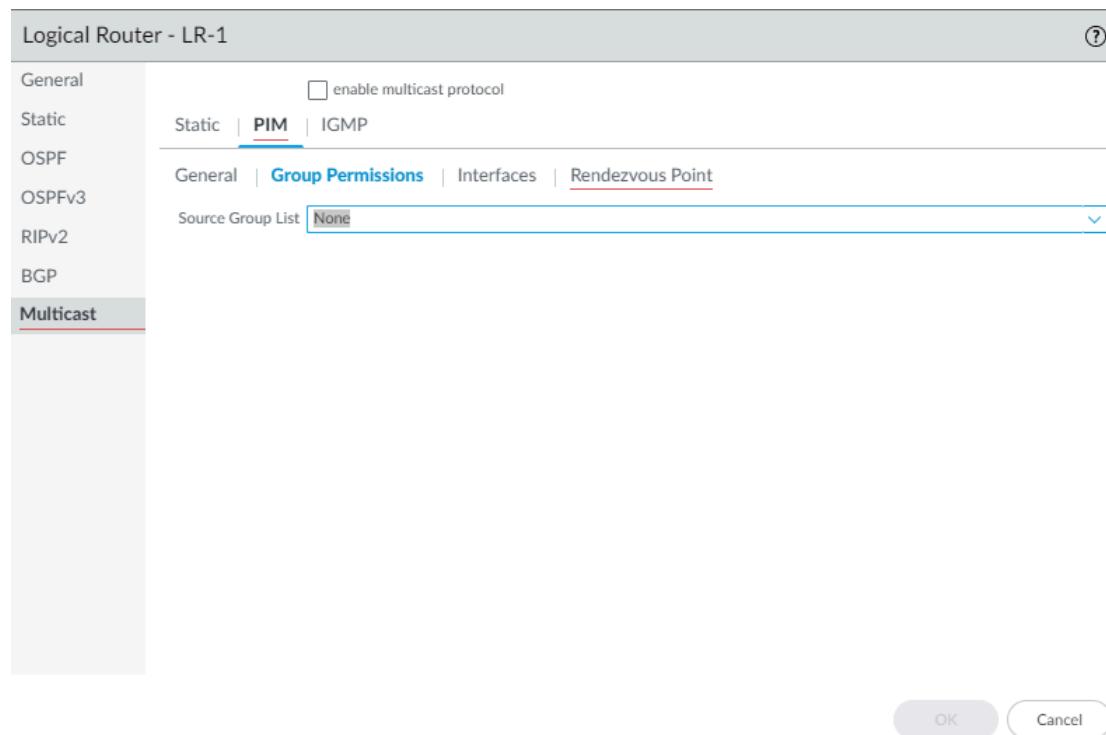
The **RPF lookup mode** also controls where to do route lookup to select the route to use for the PIM Join.

3. For the **Interface General Timer**, select a PIM Interface Timer Profile or [create a new IPv4 PIM Interface Timer profile](#); default is **None**.
4. Specify the **Route Age Out Time (sec)**—the number of seconds that a multicast route remains in the mRIB after the session ends between a multicast group and a source; range is 210 to 7,200; default is 210.
5. To configure Source-Specific Multicast (SSM), in **Multicast SSM Range** select a prefix list (or create a new one) that specifies the source addresses allowed to deliver multicast traffic to the receiver; default is **None (no prefix list)**.
6. To configure the Shortest-Path Tree (SPT) threshold for a multicast group or prefix, **Add a Group Address** (multicast group or prefix for which you are specifying the distribution tree) by selecting a [Prefix List](#) or creating a new one.
7. Specify the **Threshold** rate in kilobits per second (kbps); if multicast traffic for the multicast group/prefix arrives at the logical router faster than this threshold rate, routing to the specified group/prefix switches from shared tree (sourced from the Rendezvous Point [RP]) to SPT distribution:
 - **0 (switch on first data packet)** (default)—The logical router switches from shared tree to SPT for the group/prefix when the logical router receives the first data packet for the group/prefix.
 - Enter the total number of kilobits per second that can arrive for the multicast group/prefix at any interface and over any time period, upon which the logical router switches to SPT distribution for that multicast group or prefix; range is 0 to 4,294,967,295.
 - **never (do not switch to spt)**—The PIM router continues to use the shared tree to forward packets to the multicast group/prefix.

STEP 5 | Specify PIM group permissions to control which PIM Join messages and Register messages the logical router accepts, and which multicast traffic the logical router forwards.

1. Select **PIM > Group Permissions**.
2. To control packets to certain destination multicast groups from certain sources (S,G) to transit the logical router, for **Source Group List**, select an [Access List](#) that you created or create a new one. The access list can be an extended access list where the source

specifies the multicast source and the destination specifies the multicast group. Default is **None (no access list)**.



When you modify PIM Group Permissions by removing or changing the Source Group access list, the new permission does not retroactively clear multicast routes from the multicast RIB table (mRIB) or multicast FIB table (mFIB) for existing flows. To change entries for existing flows in the mRIB or mFIB, you would need to force a Leave or clear mroute entry.

STEP 6 | Configure PIM characteristics for an interface.

1. Select **PIM > Interfaces** and Add an interface by **Name**.

A screenshot of a configuration dialog titled "IPv4 Multicast - PIM Interface". It contains fields for Name, Description, DR Priority (set to 1), Send BSM (checked), Timer Profile (set to None), and Neighbor Filter (set to None). At the bottom are "OK" and "Cancel" buttons.

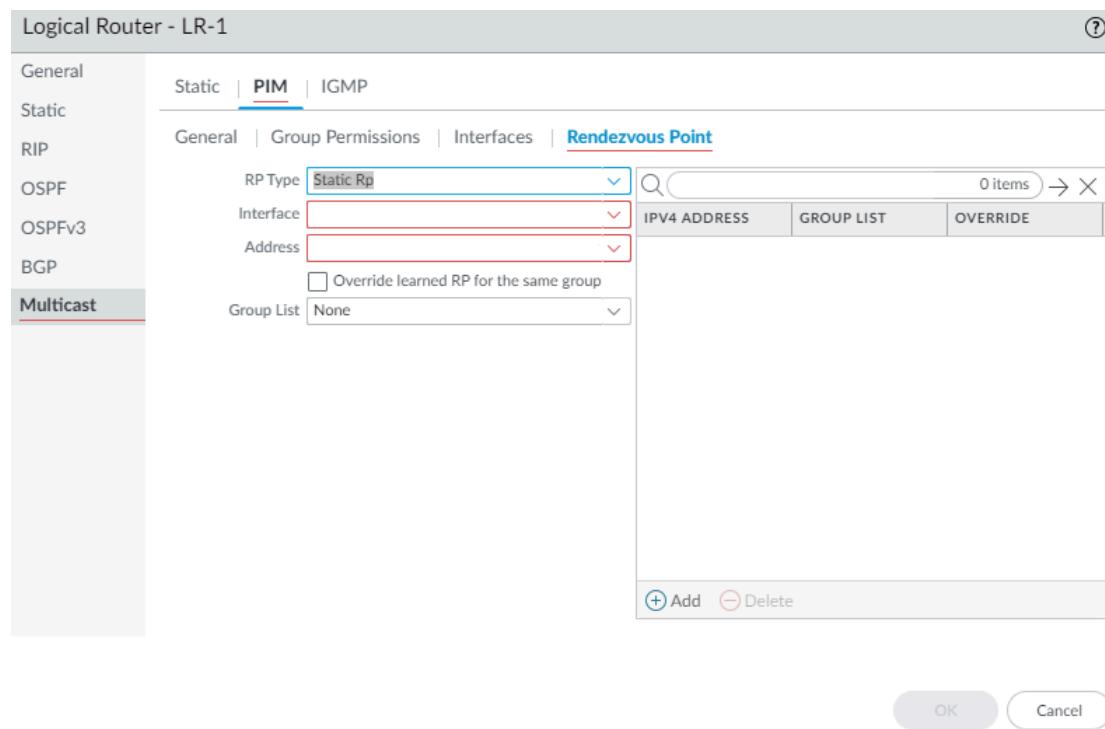
2. Enter a helpful **Description** of the interface.
3. Specify the **DR Priority** (Designated Router priority) of the interface to control which router forwards PIM Join messages, PIM Register messages, and Prune messages to the Rendezvous Point (RP); range is 1 to 4,294,967,295; default is 1. Of the PIM devices on

- a LAN, if DR Priority is configured, the device with the highest priority value is elected the DR.
4. **Send BSM** to allow propagation of Bootstrap Messages (enabled by default).
 *The Advanced Routing Engine cannot act as a BSR, but can send and relay BSM messages.*
 5. The **Timer Profile** for the interface is inherited from the General PIM section unless you override that by selecting an [IPv4 PIM Interface Timer profile](#); default is **None**.
 6. Specify a **Neighbor Filter** using an [access list](#) you created or create a new access list to specify the prefixes of devices that are allowed to become or denied from becoming PIM neighbors of the logical router.
 7. Click **OK**.

STEP 7 | (ASM only) Configure a **PIM Rendezvous Point (RP)** for an Any-Source Multicast (ASM) environment.

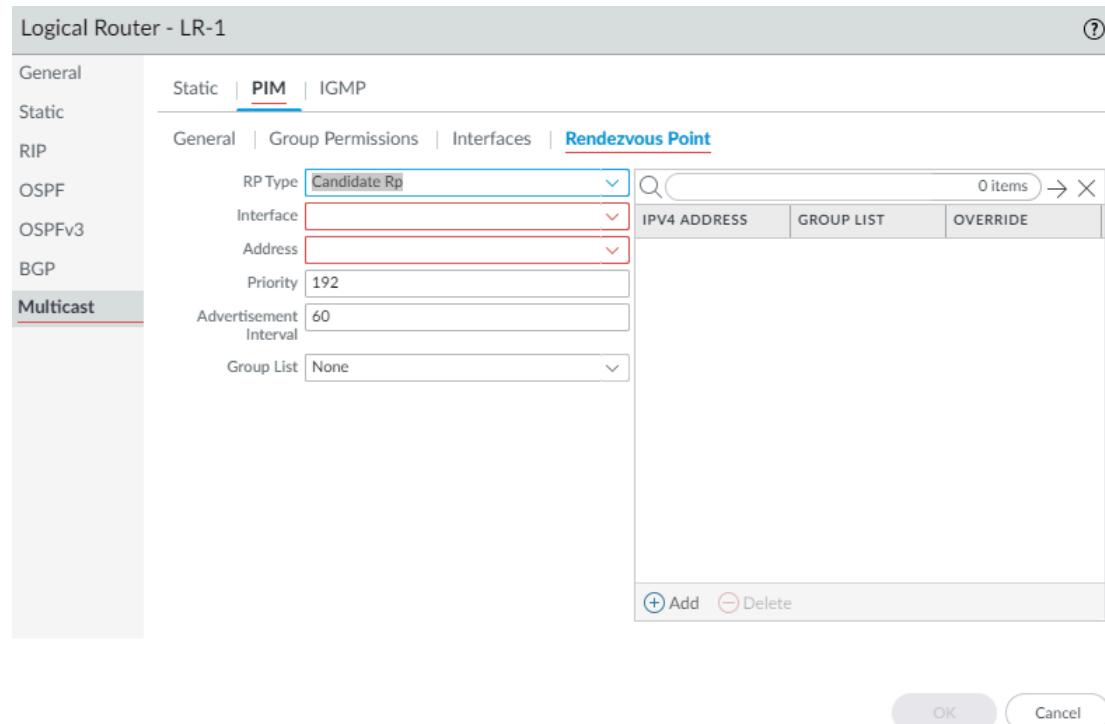
 You can configure a **Candidate RP** and a **Static RP**; they are not mutually exclusive.

1. Select **PIM > Rendezvous Point**.
2. Select the local **RP Type**: **Static RP** or **Candidate RP**; default is **None**.
3. If you choose **Static RP**, this establishes a static mapping of an RP to multicast groups. You must explicitly configure the same RP on other PIM routers in the PIM domain. Configure the following:
 - Select the **RP Interface** where the RP receives and sends multicast packets. Valid interface types are Layer3 interfaces (which include Ethernet, VLAN, loopback, Aggregate Ethernet (AE), tunnel, and subinterfaces).
 - Select the **Address** of the interface; the IP addresses of the interface you selected populate the list.
 - Select **Override learned RP for the same group** so that this static RP serves as RP instead of the RP elected for the groups in the Group List.
 - Specify the **Group List** of multicast groups for which the static RP acts as the RP by selecting an **Access List** or creating a new access list. Default is **None (no access list)**.



4. If you choose **Candidate RP**:
 - Select the **Interface** where the candidate RP receives and sends multicast packets. Valid interface types are Layer3 interfaces (which include Ethernet, VLAN, loopback, Aggregate Ethernet (AE), tunnel, and subinterfaces).
 - Select the **Address** of the interface.

- Specify the **Priority** of the candidate RP; range is 0 to 255; default is 192. A lower priority value indicates a higher priority.
- Specify the **Advertisement Interval**, the frequency (in seconds) at which the candidate RP sends advertisements to other routers; range is 1 to 26,214; default is 60.
- To control the groups that the candidate RP accepts, select a **Group List**, which is an IPv4 [access list](#) you created, or create a new access list. Default is **None (no access list)**. If no access list is applied, the logical router starts advertising itself as the RP for all groups.

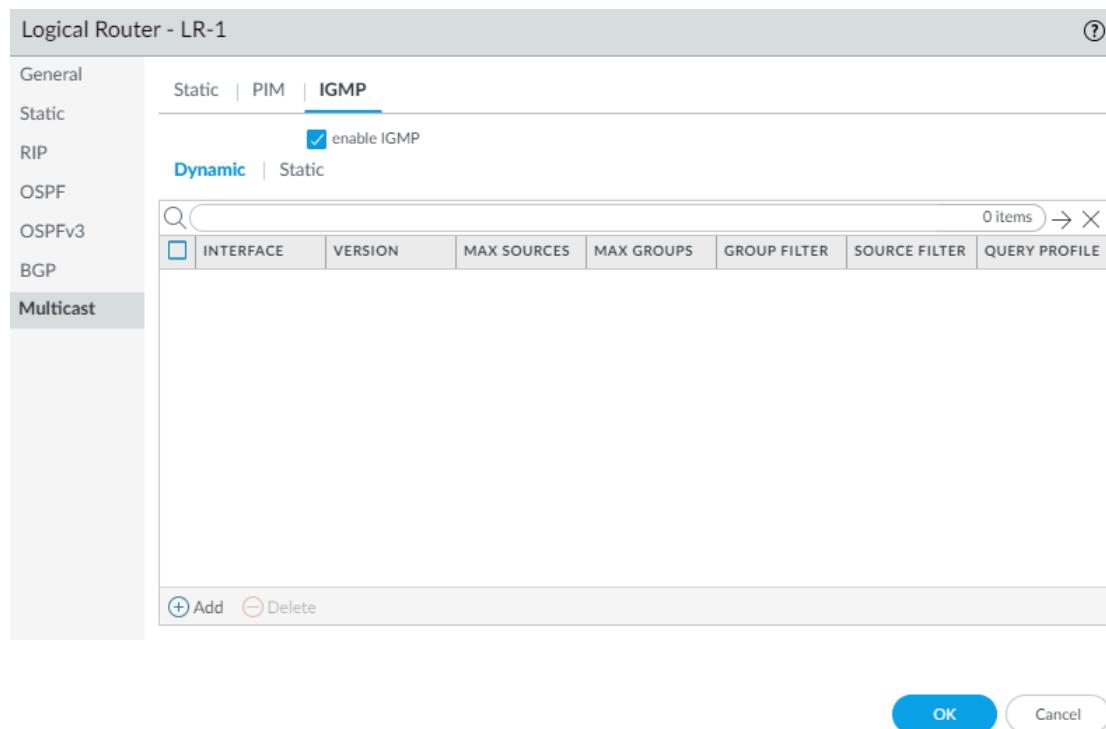


- Add an **IPv4 Address** of the remote (external) RP.
- Select a **Group List** to specify the multicast groups for which the remote RP acts as the RP or create a new access list. Default is **None (no access list)**.
- Select **Override** if you want the remote RP you statically configured to serve as RP instead of an RP that is dynamically learned (elected) for the groups in the Group List.
- Click **OK**.

STEP 8 | Click **OK** to save PIM settings.

STEP 9 | Configure IGMP on interfaces that face a multicast receiver.

1. Select **IGMP** and enable **IGMP**.



2. To configure a dynamic IGMP interface, select **Dynamic**.

1. Add an **Interface** by selecting one from the list.

IPv4 Multicast - IGMP Dynamic	
Interface	<input type="text"/>
Version	<input type="radio"/> 2 <input checked="" type="radio"/> 3
Robustness	<input type="text" value="2"/>
Group Filter	<input type="text" value="None"/>
Max Groups	<input type="text" value="unlimited"/>
Max Sources	<input type="text" value="unlimited"/>
Query Profile	<input type="text" value="None"/>
<input type="checkbox"/> drop IGMP packets without Router Alert option	
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

2. Select the **IGMP Version: 2 or 3**.

3. Select the **Robustness** value in the range 1 to 7; default is 2.



The $(\text{Robustness} * \text{QueryInterval}) + \text{MaxQueryResponseTime}$ determines how long a Join message is valid on the logical router. If the logical router receives a Leave Group message, $\text{Robustness} * \text{LastMemberQueryInterval}$ is the length of time that the logical router waits before deleting the Leave Group entry.

Increase the Robustness value if the subnet on which this logical router is located is prone to losing packets. For Join messages, a Robustness value of 1 is ignored. For Leave Group messages, the logical router uses the Robustness value as the Last Member Query Count also.

4. For **Group Filter**, select an [access list](#) or create a new access list to control the sources and groups for which the interface will accept IGMP Joins; default is **None (no access list)**.
5. For **Max Groups**, enter the maximum number of groups that IGMP can process simultaneously for the interface. Range is 1 to 65,535; default is **unlimited**, which means the highest value in the range.
6. For **Max Sources**, enter the maximum number of sources that IGMP can process simultaneously for the interface. Range is 1 to 65,535; default is **unlimited**, which means the highest value in the range.
7. For **Query Profile**, select an [IGMP Interface Query profile](#) you created or create a new one to apply to the interface; default is **None**.
8. Select **drop IGMP packets without Router Alert option** to require that incoming IGMPv2 or IGMPv3 packets have the [IP Router Alert Option](#), RFC 2113, or they will be dropped. (Default is disabled.)
9. Click **OK** to save the dynamic IGMP interface.

3. To configure a static IGMP interface, select **Static**.

- 1. Add a static interface by Name.**

Name	<input type="text"/>
Interface	<input type="text" value="None"/>
Group Address	<input type="text"/>
Source Address	<input type="text"/>
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

2. Select the **Interface** to be the static IGMP interface.
3. Enter the multicast **Group Address** of the static IGMP members.
4. Enter the **Source Address** of the sender transmitting multicast traffic to the multicast group (S,G). Traffic for this (S,G) combination is allowed on the static IGMP interface.
5. Click **OK** to save the static IGMP interface.

STEP 10 | Click **OK** to save the multicast configuration.

STEP 11 | Commit.

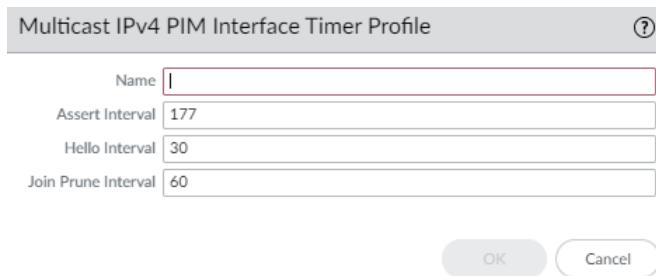
Create Multicast Routing Profiles

On an Advanced Routing Engine, create the following routing profiles to apply to an [IPv4 multicast configuration](#):

- **Multicast IPv4 PIM Interface Timer Profiles**—Use on the PIM General tab (Interface General Timer) and on the PIM Interfaces tab to override the Interface General Timer.
- **Multicast IPv4 IGMP Interface Query Profiles**—Use on the IGMP tab for a Dynamic IGMP interface.

STEP 1 | Create a multicast IPv4 PIM Interface Timer profile.

1. Select **Network > Routing > Routing Profiles > Multicast**.
2. Add a **Multicast IPv4 PIM Interface Timer Profile by Name**. (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Specify the **Assert Interval**—Number of seconds between [PIM Assert messages](#) that the logical router sends to other PIM routers on the multiaccess network when they are electing a PIM forwarder; range is 0 to 65,534; default is 177.
4. Specify the **Hello Interval**—Number of seconds between PIM Hello messages that the logical router sends to its PIM neighbors from each interface in the interface group; range is 1 to 180; default is 30.
5. Specify the **Join Prune Interval**—Number of seconds between PIM Join messages (and between PIM Prune messages) that the logical router sends upstream toward a multicast source; range is 60 to 600; default is 60.

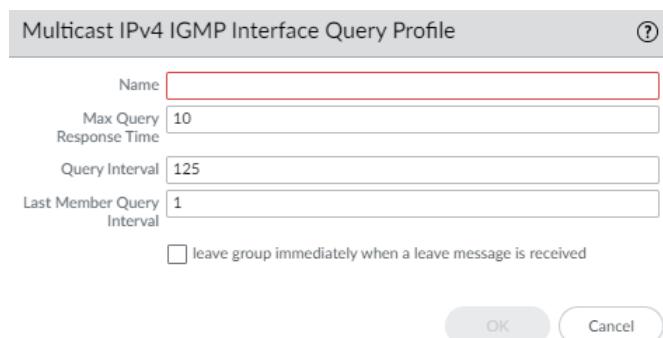


6. Click **OK**.

STEP 2 | Create a multicast IPv4 IGMP Interface Query profile.

1. Select **Network > Routing > Routing Profiles > Multicast**.
2. Add a **Multicast IPv4 IGMP Interface Query Profile by Name** (maximum of 63 characters). The name must start with an alphanumeric character, underscore (_), or hyphen (-), and can contain a combination of alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.
3. Specify the **Max Query Response Time**—maximum number of seconds allowed for a receiver to respond to an IGMP membership Query message before the logical router

- determines that the receiver no longer wants to receive multicast packets for the group; range is 1 to 25; default is 10.
4. Specify the **Query Interval**—number of seconds between IGMP membership Query messages that the logical router sends to a receiver to determine whether the receiver still wants to receive the multicast packets for a group; range is 1 to 1,800; default is 125.
 5. Specify the **Last Member Query Interval**—number of seconds allowed for a receiver to respond to a Group-Specific Query that the logical router sends after a receiver sends a Leave Group message; range is 1 to 25; default is 1.
 6. If you enable **leave group immediately when a leave message is received**, when there is only one member in a multicast group and the logical router receives an IGMP Leave message for that group, this setting causes the logical router to remove that group and outgoing interface from the multicast routing information base (mRIB) and multicast forwarding information base (mFIB) immediately, rather than waiting for the Last Member Query Interval to expire. Enabling this setting saves network resources. (Default is disabled.)



7. Click **OK**.

STEP 3 | Commit your changes.

Create an IPv4 MRoute

The Advanced Routing Engine allows you to [Configure IPv4 Multicast](#) routing for a logical router. Recall that PIM checks whether the firewall received the packets on the same interface that the firewall uses to send unicast packets back to the source, by checking the unicast RIB.

In a topology where you want unicast packets to take a different route from multicast packets, you can configure an mroute. An mroute is static unicast route that points to a multicast source; the mroute is stored in the multicast RIB (MRIB). PIM uses the mroute for the RPF checks, rather than using the unicast RIB for RPF checks. Whether PIM uses the MRIB or URIB for RPF checking depends on the RPF lookup mode configured for PIM. During RPF checks, the mroute used is the one with the longest prefix match.

An mroute is useful, for example, when some devices along the path do not support multicast routing, so a tunnel is used to connect multicast routers.

STEP 1 | Configure a Logical Router.

STEP 2 | Select **Network > Routing > Logical Routers** and select a logical router.

STEP 3 | Select **Multicast** and **enable multicast protocol**.

STEP 4 | Create an mroute.

1. Select **Static** and **Add** an mroute by **Name**. The name must start with an alphanumeric character, underscore (_), or hyphen (-), and contain zero or more alphanumeric characters, underscore, or hyphen. No dot (.) or space is allowed.

The screenshot shows a configuration dialog titled "IPv4 Multicast - Static Route". It contains the following fields:

- Name:** A text input field.
- Destination:** A dropdown menu.
- Interface:** A dropdown menu.
- Next Hop:** A dropdown menu.
- Preference:** A text input field with a range of 1 - 255.

At the bottom right are two buttons: "OK" and "Cancel".

2. Enter the **Destination** (IPv4 Address/Mask or address object) of the mroute, which is the multicast source or subnet to which the firewall performs an RPF check.
3. Select the egress **Interface** for the unicast route to the multicast source.
4. Enter the IPv4 address (or address object) of the **Next Hop** router toward the source.
5. Enter a **Preference** for the route; range is 1 to 255.
6. Click **OK**.

STEP 5 | Click **OK**.

STEP 6 | Commit your changes.

