Security Configuration Guide: Zone-Based Policy Firewall Cisco IOS XE Release 2

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

Zone-Based Policy Firewall

This module describes the Cisco IOS XE unidirectional firewall policy between groups of interfaces known as zones.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Zone-Based Policy Firewall

The general guideline before you create zones is that you should group interfaces that are similar when they are viewed from a security perspective.

Restrictions for Zone-Based Policy Firewall

- Application-level maps (also referred to as Layer 7 class maps) are not supported in Cisco IOS XE software.
- In a Cisco Wide Area Application Services (WAAS) and Cisco IOS XE firewall configuration, all packets processed by a Wide Area Application Engine (WAE) device must go over the Cisco IOS XE firewall in both directions to support the Web Cache Coordination Protocol (WCCP) generic routing encapsulation (GRE) redirect. This situation occurs when the Layer 2 redirect is not available. If Layer 2 redirect is configured on the WAE, the system defaults to the GRE redirect to continue to function.

- When an in-to-out zone-based policy is configured to match the Internet Control Message Protocol (ICMP) on a Windows system, the **traceroute** command works. However, the same configuration on an Apple system does not work because it uses a UDP-based traceroute. To overcome this issue, configure an out-to-in zone-based policy with the **icmp time-exceeded** and **icmp host unreachable** commands with the **pass** command (not the **inspect** command).
- In a WAAS and Cisco IOS XE firewall configuration, WCCP does not support traffic redirection using policy-based routing (PBR).
- Stateful inspection support for multicast traffic is not supported between any zones, including the self zone. Use Control Plane Policing for protection of the control plane against multicast traffic.
- A UDP-based traceroute is not supported through ICMP inspection.

Information About Zone-Based Policy Firewall

Top-level Class Maps and Policy Maps

Top-level class maps allow you to identify the traffic stream at a high level. This is accomplished by using the **match access-group** and **match protocol** commands. Top-level class maps are also referred to as Layer 3 and Layer 4 class maps.

Top-level policy maps allow you to define high-level actions by using the **inspect**, **drop**, and **pass** commands. You can attach policy maps to a target (zone pair).

Note Only inspect type policies can be configured on a zone pair.

Overview of Zones

A zone is a group of interfaces that have similar functions or features. They help you specify where a Cisco IOS XE firewall should be applied.

For example, on a router, the Gigabit Ethernet interface 0/0/0 and the Gigabit Ethernet interface 0/0/1 may be connected to the local LAN. These two interfaces are similar because they represent the internal network, so they can be grouped into a zone for firewall configurations.

By default, the traffic between interfaces in the same zone is not subject to any firewall policy and traffic passes freely between the interfaces. Firewall zones are used for security.

Note Zones may not span interfaces in different VPN routing and forwarding (VRF) instances.

Security Zones

A security zone is a group of interfaces to which a policy can be applied.

Grouping interfaces into zones involves the following two procedures:

- Creating a zone so that interfaces can be attached to it.
- Configuring an interface to be a member of a given zone.

By default, traffic flows among interfaces that are members of the same zone.

When an interface is a member of a security zone, all traffic to and from that interface (except traffic going to the router or initiated by the router) is dropped. To permit traffic to and from a zone-member interface, you must make that zone part of a zone pair and then apply a policy to that zone pair. If the policy permits traffic (through **inspect** or **pass** actions), traffic can flow through the interface.

Basic rules to consider when setting up zones are as follows:

- Traffic from a zone interface to a nonzone interface or from a nonzone interface to a zone interface is always dropped.
- Traffic between two zone interfaces is inspected if there is a zone pair relationship for each zone and if there is a configured policy for that zone pair.
- By default, all traffic between two interfaces in the same zone is always allowed as if the **pass** action is configured.
- A zone pair can be configured with a zone as both the source and the destination zone. An inspect policy can be configured on this zone pair to inspect or drop the traffic between two interfaces in the same zone.

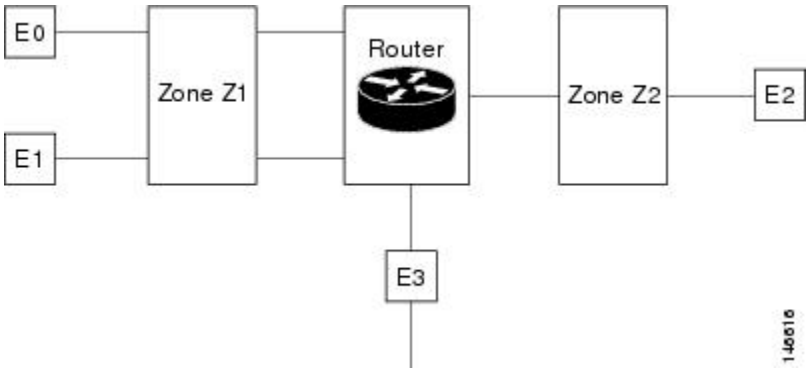
For traffic to flow among all interfaces in a router, these interfaces must be a member of a security zone.

It is not necessary for all router interfaces to be members of security zones.

The figure below illustrates the following:

- Interfaces E0 and E1 are members of security zone Z1.
- Interface E2 is a member of security zone Z2.
- Interface E3 is not a member of any security zone.

Figure 1 Security Zone Restrictions



The following situations exist:

- The zone pair and policy are configured in the same zone. Traffic flows freely between interfaces E0 and E1 because they are members of the same security zone (Z1).
- If no policies are configured, traffic will not flow between any other interfaces (for example, E0 and E2, E1 and E2, E3 and E1, and E3 and E2).
- Traffic can flow between E0 or E1 and E2 only when an explicit policy permitting traffic is configured between zone Z1 and zone Z2.
- Traffic can never flow between E3 and E0/E1/E2 unless default zones are enabled.

Overview of Security Zone Firewall Policies

A class identifies a set of packets based on its contents. Normally, you define a class so that you can apply an action on the identified traffic that reflects a policy. A class is designated through class maps.

An action is a functionality that is typically associated with a traffic class. For example, **inspect**, **drop**, and **pass** are actions.

To create firewall policies, you must complete the following tasks:

- Define match criteria (class map)
- Associate actions to the match criteria (policy map)
- Attach the policy map to a zone pair (service policy)

The **class-map** command creates a class map to be used for matching packets to a specified class. Packets arriving at the targets (such as the input interface, output interface, or zone pair), determined by how the **service-policy** command is configured, are checked against the match criteria configured for a class map to determine if the packet belongs to that class.

The **policy-map** command creates or modifies a policy map that can be attached to one or more targets to specify a service policy. Use the **policy-map** command to specify the name of the policy map to be created, added to, or modified before you can configure policies for classes whose match criteria are defined in a class map.

Virtual Interfaces as Members of Security Zones

A virtual interface is a logical interface configured with generic configuration information for a specific purpose or for configuration common to specific users, plus router-dependent information. The template contains Cisco IOS XE software interface commands that are applied to virtual access interfaces, as needed. To configure a virtual template interface, use the **interface virtual-template** command.

Zone member information is acquired from a RADIUS server and then the dynamically created virtual interface is made a member of that zone. The **zone-member security** command puts the interface into the corresponding zone.

For more information on Per Subscriber Firewall on LNS feature, see the *Release Notes for Cisco ASR 1000 Series Aggregation Services Routers for Cisco IOS XE Release 2*.

Zone Pairs

A zone pair allows you to specify a unidirectional firewall policy between two security zones.

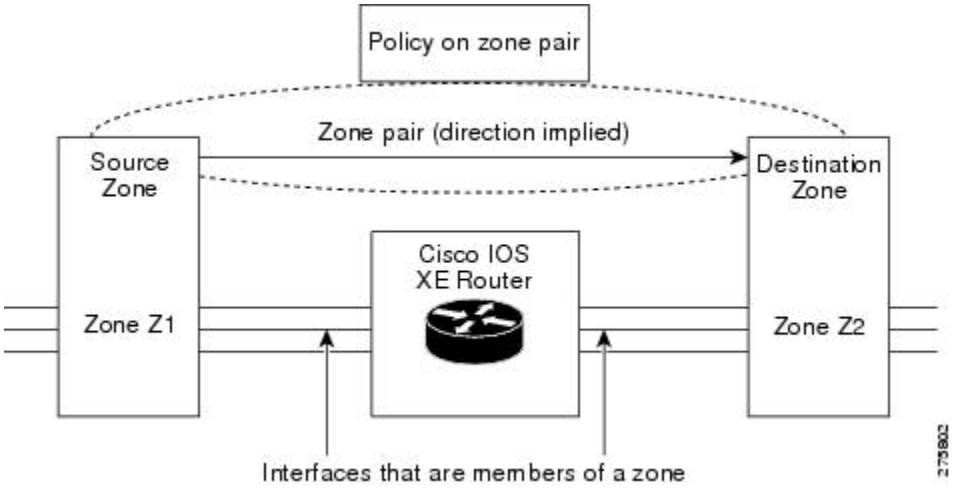
To define a zone pair, use the **zone-pair security** command. The direction of the traffic is specified by configuring a source and a destination zone. The source and destination zones of a zone pair must be security zones.

You can select the default or self zone as either the source or the destination zone. The self zone is a system-defined zone. It does not have any interfaces as members. A zone pair that includes the self zone, along with the associated policy, is applied to traffic directed to the router or traffic generated by the router. It does not apply to traffic through the router. The most common use of firewalls is to apply them to traffic through a router, so you need at least two zones (that is, you cannot use the self zone).

To permit traffic between zone member interfaces, you must configure a policy that permits (or inspects) traffic between that zone and another zone. To attach a firewall policy map to the target zone pair, use the **service-policy type inspect** command.

The figure below shows the application of a firewall policy to traffic flowing from zone Z1 to zone Z2, which means that the ingress interface for the traffic is a member of zone Z1 and the egress interface is a member of zone Z2.

Figure 2 Zone Pairs



If there are two zones and you require policies for traffic going in both directions (from Z1 to Z2 and Z2 to Z1), you must configure two zone pairs (one for each direction).

If a policy is not configured between a pair of zones, traffic is dropped. However, it is not necessary to configure a zone pair and a service policy solely for the return traffic. By default, return traffic is not allowed. If a service policy inspects the traffic in the forward direction and there is no zone pair and service policy for the return traffic, the return traffic is inspected. If a service policy passes the traffic in the forward direction and there is no zone pair and service policy for the return traffic, the return traffic is dropped. In both these cases, you need to configure a zone pair and a service policy to allow the return traffic. In the above figure, it is not mandatory that you configure a zone pair source and destination solely for allowing return traffic from Z2 to Z1. The service policy on the Z1 to Z2 zone pair takes care of it.

Zones and Inspection

Zone-based policy firewalls examine the source and destination zones from the ingress and egress interfaces for a firewall policy. It is not necessary that all traffic flowing to or from an interface be inspected; you can designate that individual flows in a zone pair be inspected through your policy map that you apply across the zone pair. The policy map will contain class maps that specify the individual flows.

You can also configure **inspect** parameters like TCP thresholds and timeouts on a per-flow basis.

Zones and ACLs

ACLs applied to interfaces that are members of zones are processed before the policy is applied on the zone pair. You must make sure that interface ACLs do not interfere with the policy firewall traffic when there are policies between zones.

Pinholes (are ports opened through a firewall that allows applications controlled access to a protected network) are not punched for return traffic in interface Access Control Lists (ACLs).

Class Maps and Policy Maps for Zone-Based Policy Firewalls

Quality of service (QoS) class maps have numerous match criteria; firewalls have fewer match criteria. Firewall class maps have the type inspect and this information controls what shows up under firewall class maps.

A policy is an association of traffic classes and actions. It specifies what actions should be performed on defined traffic classes. An action is a specific function, and it is typically associated with a traffic class. For example, **inspect** and **drop** are actions.

Layer 3 and Layer 4 Class Maps and Policy Maps

Layer 3 and Layer 4 class maps identify traffic streams on which different actions are performed.

A Layer 3 or Layer 4 policy map is sufficient for the basic inspection of traffic.

The following example shows how to configure class map c1 with the match criteria of ACL 101 and the FTP protocol, and create an inspect policy map named p1 to specify that packets will be dropped on the traffic at c1:

```
Router(config)# class-map type inspect match-all c1
Router(config-cmap)# match access-group 101
Router(config-cmap)# match protocol ftp
Router(config-cmap)# exit
Router(config)# policy-map type inspect p1
Router(config-pmap)# class type inspect c1
Router(config-pmap-c)# drop
```


Supported Protocols

The following protocols are supported:

- FTP
- H.323
- ICMP
- Lightweight Directory Access Protocol (LDAP)
- LDAP over Transport Layer Security/Secure Socket Layer (LDAPS)
- Real-time Streaming Protocol (RTSP)
- Session Initiation Protocol (SIP)

- SCCP (Skinny Client Control Protocol)
- TCP
- TFTP
- UDP

Class-Map Configuration Restriction

If a traffic meets multiple match criteria, these match criteria must be applied in the order of specific to less specific. For example, consider the following class map example:

```
class-map type inspect match-any my-test-cmap
 match protocol ftp
 match protocol tcp
```

In this example, the FTP traffic must first encounter the **match protocol ftp** command to ensure that the traffic will be handled by the service-specific capabilities of FTP inspection. If the “match” lines were reversed so that the traffic encountered the **match protocol tcp** command before it was compared to the **match protocol ftp** command, the traffic would simply be classified as TCP traffic and inspected according to the capabilities of the firewall’s TCP Inspection component.

Class-Default Class Map

In addition to user-defined classes, a system-defined class map named class-default represents all packets that do not match any of the user-defined classes in a policy. It always is the last class in a policy map.

You can define explicit actions for this group of packets. If you do not configure any actions for class-default in an inspect policy, the default action is **drop**.

Access Control List and Class Map

Access lists are packet-classifying mechanisms. Access lists define the actual network traffic that is permitted or denied when an ACL is applied to a particular router network interface. Thus, the ACL is a sequential collection of permit and deny conditions that applies to a packet. A router tests packets against the conditions set in the ACL one at a time. A deny condition is interpreted as “do not match.” Packets that match a deny access control entry (ACE) cause an ACL process to terminate and the next match statement within the class to be examined.

Class maps are used to match a range of variables in an ACL based on the following criteria:

- If a class map does not match a permit or a deny condition, then the ACL fails.
- If a class map is specified, the class map performs either an AND (match-all) or an OR (match-any) operation on the ACL variables.
- If a match-all attribute is specified and any match condition, ACL, or protocol fails to match the packet, further evaluation of the current class is stopped, and the next class in the policy is examined.
- If any match in a match-any attribute succeeds, the class map criteria are met and the action defined in the policy is performed.
- If an ACL matches the match-any attribute, the firewall attempts to ascertain the Layer 7 protocol based on the destination port.

If you specify the match-all attribute in a class map, the Layer 4 match criteria (ICMP, TCP, and UDP) are set and the Layer 7 match criteria are not set. Hence, the Layer 4 inspection is performed and Layer 7 inspection is omitted.

Access lists come in different forms: standard and extended access lists. Standard access lists are defined to permit or deny an IP address or a range of IP addresses. Extended access lists define both the source and

the destination IP address or an IP address range. Extended access lists can also be defined to permit or deny packets based on ICMP, TCP, and UDP protocol types and the destination port number of the packet.

The following example shows how a packet received from the IP address 10.2.3.4 is matched with the class test1. In this example, the access list 102 matches the deny condition and stops processing other entries in the access list. Because the class map is specified with a match-all attribute, the “class-map test1” match fails. However, the class map is inspected if it matches one of the protocols listed in test1 class map.

If the class map test1 had a match-any attribute (instead of match-all), then the ACL would have matched deny and failed, but then the ACL would have matched the HTTP protocol and performed the inspection using “pmap1.”

```
access-list 102 deny ip 10.2.3.4 0.0.0. any
access-list 102 permit any any
class-map type inspect match-all test1
match access-list 102
match protocol http
class-map type inspect match-any test2
 match protocol sip
 match protocol ftp
 match protocol http
parameter-map type inspect pmap1
 tcp idle-time 15
parameter-map type inspect pmap2
 udp idle-time 3600
policy-map type inspect test
 class type inspect test1
 inspect pmap1
 class type inspect test2
 inspect pmap2
class type inspect class-default
 drop log
```

Firewall and Network Address Translation

Network Address Translation (NAT) enables private IP internetworks that use nonregistered IP addresses to connect to the Internet. NAT operates on a router, usually connecting two networks, and translates the private (not globally unique) addresses in the internal network into legal addresses before packets are forwarded onto another network. NAT can be configured to advertise only one address for the entire network to the outside world. A router configured with NAT will have at least one interface to the inside network and one to the outside network.

In a typical environment, NAT is configured at the exit router between a stub domain and the backbone. When a packet leaves the domain, NAT translates the locally significant source address into a globally unique address. When a packet enters the domain, NAT translates the globally unique destination address into a local address. If more than one exit point exists, each NAT must have the same translation table. If the software cannot allocate an address because it has run out of addresses, it drops the packet and sends an ICMP host unreachable packet.

With reference to NAT, the term “inside” refers to those networks that are owned by an organization and that must be translated. Inside this domain, hosts will have addresses in one address space. When NAT is configured and when the hosts are outside, hosts will appear to have addresses in another address space. The inside address space is referred to as the local address space and the outside address space is referred to as the global address space.

Consider a scenario where NAT translates both the source and the destination IP addresses. A packet is sent to a router from inside NAT with the source address 192.168.1.1 and the destination address 10.1.1.1. NAT translates these addresses and sends the packet to the external network with the source address 209.165.200.225 and the destination address 209.165.200.224.

Similarly, when the response comes back from outside NAT, the source address will be 209.165.200.225 and the destination address will be 209.165.200.224. Therefore, inside NAT, the packets will have a source address of 10.1.1.1 and a destination address of 192.168.1.1.

In this scenario, if you want to create an Application Control Engine (ACE) to be used in a firewall policy, the pre-NAT IP addresses (also known as inside local and outside global addresses) 192.168.1.1 and 209.165.200.224 must be used.

WAAS and Firewall Integration Support

The WAAS software optimizes security-compliant WANs and application acceleration solutions with the following benefits:

- Optimizes a WAN through full stateful inspection capabilities.
- Simplifies Payment Card Industry (PCI) compliance.
- Protects transparent WAN accelerated traffic.
- Integrates WAAS networks transparently.
- Supports the Network Management Equipment (NME) WAE modules or standalone WAAS device deployment.

WAAS has an automatic discovery mechanism that uses TCP options during the initial three-way handshake used to identify WAE devices transparently. After automatic discovery, optimized traffic flows (paths) experience a change in the TCP sequence number to allow endpoints to distinguish between optimized and nonoptimized traffic flows.

Note Paths are synonymous with connections.

The Cisco IOS XE firewall automatically discovers WAAS optimized traffic by enabling the sequence number to change without compromising the stateful Layer 4 inspection of TCP traffic flows that contain internal firewall TCP state variables. These variables are adjusted for the presence of WAE devices.

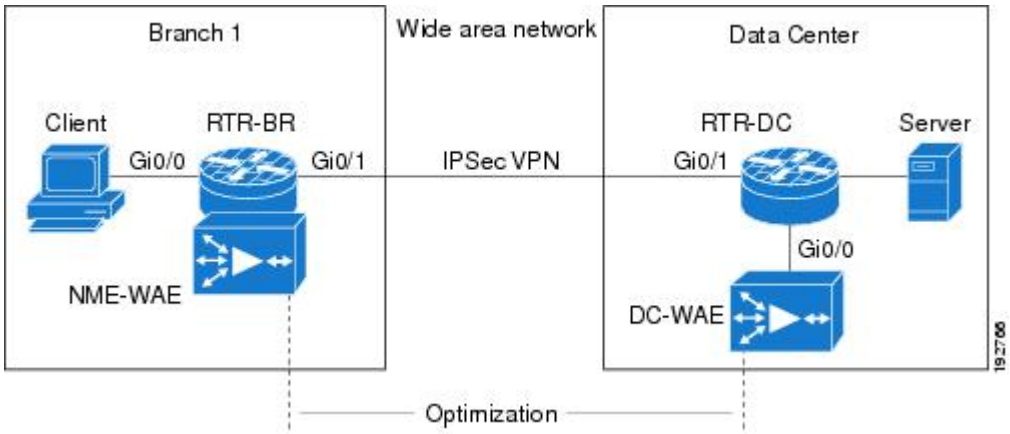
If the Cisco IOS XE firewall notices that a traffic flow has successfully completed WAAS automatic discovery, it permits the initial sequence number shift for the traffic flow and maintains the Layer 4 state on the optimized traffic flow.

WAAS Traffic Flow Optimization Deployment Scenarios

The following sections describe three different WAAS traffic flow optimization scenarios for branch office deployments. WAAS traffic flow optimization works with the Cisco IOS XE firewall feature on Cisco Aggregation Services Routers (ASRs).

The figure below shows an example of an end-to-end WAAS traffic flow optimization with the Cisco IOS XE firewall. In this particular deployment, an NME-WAE device is on the Cisco IOS Integrated Services Router (ISR). WCCP is used to redirect traffic for interception.

Figure 3 End-to-End WAAS Optimization Path



Note NME-WAE is not supported on ASR. Therefore, to support NME-WAE in the branch office must be an ISR.

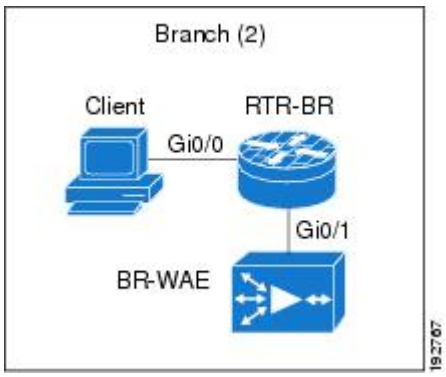
- WAAS Branch Deployment with an Off-Path Device, page 10
- WAAS Branch Deployment with an Inline Device, page 11

WAAS Branch Deployment with an Off-Path Device

A WAE device can be either an NME-WAE that is installed on an ISR as an integrated service engine or a standalone WAE device.

The figure below shows a WAAS branch deployment that uses WCCP to redirect traffic to an off-path, standalone WAE device for traffic interception. The configuration for this option is the same as the WAAS branch deployment with an NME-WAE.

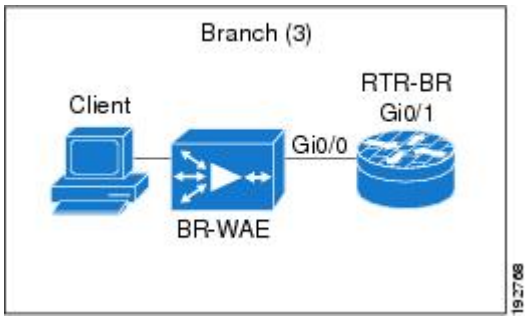
Figure 4 WAAS Off-Path Branch Deployment



WAAS Branch Deployment with an Inline Device

The figure below shows a WAAS branch deployment that has an inline WAE device that is physically in front of the router. Because the WAE device is in front of the router, Layer 7 inspection on the client side is not supported because the Cisco IOS XE firewall receives WAAS optimized packets.

Figure 5 WAAS Inline Path Branch Deployment



An edge WAAS device with the Cisco IOS XE firewall is applied at branch office sites that must inspect traffic moving to and from a WAN connection. The Cisco IOS XE firewall monitors traffic for optimization indicators (TCP options and subsequent TCP sequence number changes) and allows optimized traffic to pass while still applying Layer 4 stateful inspection and deep packet inspection to all traffic, maintaining security while accommodating WAAS optimization advantages.

Note If the WAE device is in the inline location, the device enters its bypass mode after the automatic discovery process. Although the router is not directly involved in WAAS optimization, the router must be aware that WAAS optimization is applied to the traffic in order to apply the Cisco IOS XE firewall inspection to network traffic and make allowances for optimization activity if optimization indicators are present.

Out-of-Order Packet Handling in Zone-Based Policy Firewall

By default, the Cisco IOS XE firewall drops all out-of-order (OoO) packets when Layer 7 deep packet inspection (DPI) is enabled or when Layer 4 inspection with Layer 7 protocol match is enabled. Dropping out-of-order packets can cause significant delays in end applications because packets are dropped only after the retransmission timer expires (on behalf of the sender). Layer 7 inspection is a stateful packet inspection and it does not work when TCP packets are out of order.

In Cisco IOS XE Release 3.5S, if a session does not require DPI, OoO packets are allowed to pass through the router and reach their destination. All Layer 4 traffic with OoO packets are allowed to pass through to their destination. However, if a session requires Layer 7 inspection, OoO packets are still dropped. By not dropping OoO packets when DPI is not required, the need to retransmit dropped packets and the bandwidth needed to retransmit on the network is reduced.

How to Configure Zone-Based Policy Firewall

- Configuring Layer 3 and Layer 4 Firewall Policies, page 12
- Configuring an Inspect Parameter Map, page 15

Configuring Layer 3 and Layer 4 Firewall Policies

Layer 3 and Layer 4 policies are “top level” policies that are attached to the target (zone pair). Use the following tasks to configure Layer 3 and Layer 4 firewall policies:

Configuring a Class Map for a Layer 3 or Layer 4 Firewall Policy

Perform the following task to configure a class map for classifying network traffic.

Note You must perform at least one step from Step 4, 5, or 6.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map type inspect** [**match-any** | **match-all**] *class-map-name*
4. **match access-group** {*access-group* | **name** *access-group-name*}
5. **match protocol** *protocol-name*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**class-map type inspect** [**match-any** | **match-all**] *class-map-name* **Example:** `Router(config)# class-map type inspect match-all c1`	Creates a Layer 3 or Layer 4 inspect type class map. • Enters QoS class map configuration mode.
Step 4	**match access-group** {*access-group* | **name** *access-group-name*} **Example:** `Router(config-cmap)# match access-group 101`	Configures the match criteria for a class map based on the ACL name or number.
Step 5	**match protocol** *protocol-name* **Example:** `Router(config-cmap)# match protocol ftp`	Configures the match criteria for a class map based on a specified protocol.
Step 6	**end** **Example:** `Router(config-cmap)# end`	Exits QoS class map configuration mode and enters privileged EXEC mode.

Creating a Policy Map for a Layer 3 or Layer 4 Firewall Policy

Perform the following task to create a policy map for a Layer 3 or Layer 4 firewall policy that will be attached to zone pairs.

If you are creating an inspect type policy map, note that only the following actions are allowed: drop, inspect, and pass.

Note You must perform at least one step from Step 5, 6, or 7.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map type inspect** *policy-map-name*
4. **class type inspect** *class-name*
5. **inspect** [*parameter-map-name*]
6. **drop** [**log**]
7. **pass** [**log**]
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map type inspect** *policy-map-name* **Example:** `Router(config)# policy-map type inspect p1`	Creates a Layer 3 or Layer 4 inspect type policy map. • Enters policy map configuration mode.
Step 4	**class type inspect** *class-name* **Example:** `Router(config-pmap)# class type inspect c1`	Specifies the traffic (class) on which an action is to be performed and enters policy-map class configuration mode.
Step 5	**inspect** [*parameter-map-name*] **Example:** `Router(config-pmap-c)# inspect inspect-params`	Enables Cisco IOS XE stateful packet inspection.

	Command or Action	Purpose
Step 6	**drop** [**log**] **Example:** `Router(config-pmap-c)# drop`	(Optional) Drops packets that matches a defined class. **Note** The actions drop and pass are exclusive, and the actions inspect and drop are exclusive; that is, you cannot specify both of them.
Step 7	**pass** [**log**] **Example:** `Router(config-pmap-c)# pass`	(Optional) Allows packets that match a defined class.
Step 8	**end** **Example:** `Router(config-pmap-c)# end`	Exits policy-map class configuration mode and enters privileged EXEC mode.

Configuring an Inspect Parameter Map

An inspect parameter map is optional if you are using an inspect type policy. If you do not configure a parameter map, the firewall uses default parameters. Parameters associated with the inspect action apply to all nested actions (if any). If parameters are specified in both the top and lower levels, those in the lower levels override those in the top levels.

Changes to the parameter map are not reflected on connections already established through the firewall. Changes are applicable only to new connections permitted through the firewall. To ensure that your firewall enforces policies strictly, clear all connections in the firewall after you change the parameter map. To clear existing connections, use the **clear zone-pair inspect sessions** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type inspect** *parameter-map-name*
4. **alert** {**on** | **off**}
5. **audit-trail** {**on** | **off**}
6. **dns-timeout** *seconds*
7. **icmp idle-time** *seconds*
8. **max-incomplete** {**low** | **high**} *number-of-connections*
9. **one-minute** {**low** | **high**} *number-of-connections*
10. **sessions maximum** *sessions*
11. **tcp finwait-time** *seconds*
12. **tcp idle-time** *seconds*
13. **tcp max-incomplete host** *threshold* [**block-time** *minutes*]
14. **tcp synwait-time** *seconds*
15. **udp idle-time** *seconds*
16. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**parameter-map type inspect** *parameter-map-name* **Example:** `Router(config)# parameter-map type inspect eng-network-profile`	Configures an inspect parameter map for connecting thresholds, timeouts, and other parameters pertaining to the **inspect** keyword. • Enters parameter-map type inspect configuration mode.

	Command or Action	Purpose
Step 4	**alert** {**on** | **off**} **Example:** `Router(config-profile)# alert on`	(Optional) Turns on stateful packet inspection alert messages that are displayed on the console.
Step 5	**audit-trail** {**on** | **off**} **Example:** `Router(config-profile)# audit-trail on`	(Optional) Turns on audit trail messages.
Step 6	**dns-timeout** *seconds* **Example:** `Router(config-profile)# dns-timeout 60`	(Optional) Specifies the DNS idle timeout (the length of time for which a DNS lookup session will continue to be managed while there is no activity).
Step 7	**icmp idle-time** *seconds* **Example:** `Router(config-profile)# icmp idle-timeout 90`	(Optional) Configures the timeout for ICMP sessions.
Step 8	**max-incomplete** {**low** | **high**} *number-of-connections* **Example:** `Router(config-profile)# max-incomplete low 800`	(Optional) Defines the number of existing half-opened sessions that will cause the Cisco IOS XE firewall to start and stop deleting half-opened sessions.
Step 9	**one-minute** {**low** | **high**} *number-of-connections* **Example:** `Router(config-profile)# one-minute low 300`	(Optional) Defines the number of new unestablished sessions that will cause the system to start and stop deleting half-opened sessions.
Step 10	**sessions maximum** *sessions* **Example:** `Router(config-profile)# sessions maximum 200`	(Optional) Sets the maximum number of allowed sessions for the class it is associated with. • *sessions*—Maximum number of allowed sessions. Range: 1 to 2147483647.

	Command or Action	Purpose
Step 11	**tcp finwait-time** *seconds* **Example:** `Router(config-profile)# tcp finwait-time 5`	(Optional) Specifies how long a TCP session will be managed after the firewall detects a finish (FIN)-exchange.
Step 12	**tcp idle-time** *seconds* **Example:** `Router(config-profile)# tcp idle-time 90`	(Optional) Configures the timeout for TCP sessions.
Step 13	**tcp max-incomplete host** *threshold* [**block-time** *minutes*] **Example:** `Router(config-profile)# tcp max-incomplete host 500 block-time 10`	(Optional) Specifies threshold and blocking time values for TCP host-specific denial of service (DoS) detection and prevention.
Step 14	**tcp synwait-time** *seconds* **Example:** `Router(config-profile)# tcp synwait-time 3`	(Optional) Specifies how long the software will wait for a TCP session to reach the established state before dropping the session.
Step 15	**udp idle-time** *seconds* **Example:** `Router(config-profile)# udp idle-time 75`	(Optional) Configures the idle timeout of UDP sessions going through the firewall.
Step 16	**end** **Example:** `Router(config-profile)# end`	Exits parameter-map type inspect configuration mode and enters privileged EXEC mode.

Configuring NetFlow Event Logging

Global parameter maps are used for NetFlow event logging. With NetFlow event logging enabled, logs are sent to an off-box, high-speed log collector. By default, this functionality is not enabled. (If this functionality is not enabled, firewall logs are sent to a logger buffer located in the Route Processor or console.)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type inspect global**
4. **log dropped-packets**
5. **log flow-export v9 udp destination** *ipv4-address port*
6. **log flow-export template timeout-rate** *seconds*
7. **end**
8. **show parameter-map type inspect global**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**parameter-map type inspect global** **Example:** `Router(config)# parameter-map type inspect global`	Configures a global parameter map. • Enters parameter-map type inspect configuration mode.
Step 4	**log dropped-packets** **Example:** `Router(config-profile)# log dropped-packets`	Enables dropped packet logging.
Step 5	**log flow-export v9 udp destination** *ipv4-address port* **Example:** `Router(config-profile)# log flow-export v9 udp destination 192.0.2.0 5000`	Enables NetFlow event logging and provides the collector's IP address and port.

	Command or Action	Purpose
Step 6	**log flow-export template timeout-rate** *seconds* **Example:** `Router(config-profile)# log flow-export template timeout-rate 5000`	Specifies the template timeout value.
Step 7	**end** **Example:** `Router(config)# end`	Exits global configuration mode and enters privileged EXEC mode.
Step 8	**show parameter-map type inspect global** **Example:** `Router# show parameter-map type inspect global`	Displays logging configurations.

Creating Security Zones and a Zone Pair and Attaching a Policy Map to a Zone Pair

You need two security zones to create a zone pair. However, you can create only one security zone and use a system-defined security zone called “self.” Note that if you select a self zone, you cannot configure inspect policing.

Use this process to complete the following tasks:

- Create at least one security zone.
- Define zone pairs.
- Assign interfaces to security zones.
- Attach a policy map to a zone pair.

Tip The general guideline for creating a zone is that you should group interfaces that are similar when they are viewed from a security perspective.

Note

- An interface can be a member of only one security zone.
- When an interface is a member of a security zone, all traffic to and from that interface is blocked unless you configure an explicit interzone policy on a zone pair involving that zone.
- Traffic cannot flow between an interface that is a member of a security zone and an interface that is not a member of a security zone.
- For traffic to flow among all interfaces in a router, the interfaces must be members of a security zone. This is important because after you make an interface a member of a security zone, a policy action (such as inspect or pass) is explicitly allowed through the interface and packets are dropped.
- If an interface cannot be part of a security zone or a firewall policy, you may have to add that interface in a security zone and configure a “pass all” policy (that is, a “dummy” policy) between that zone and other zones to which a traffic flow is desired.
- An ACL on an interface that is a zone member should not be restrictive (strict).
- Traffic between interfaces in the same security zone is not subject to any policy; the traffic passes freely. If you have created only one zone, you can use the system-defined default zone (self) as part of a zone pair. The zone pair and its associated policy applies to the traffic directed to the router or generated by the router.
- You can use the **default** keyword to include all interfaces that are not configured on any of the security zones. In the default zone, the policy can be defined either as a source zone or destination zone.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **zone security** {*zone-name* | **default**}
4. **description** *line-of-description*
5. **exit**
6. **zone-pair security** *zone-pair-name* [**source** {*source-zone-name* | **self** | **default**} **destination** [*destination-zone-name* | **self** | **default**]]
7. **description** *line-of-description*
8. **service-policy type inspect** *policy-map-name*
9. **exit**
10. **interface** *type number*
11. **zone-member security** *zone-name*
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**zone security** {*zone-name* | **default**} **Example:** `Router(config)# zone security zone1`	Creates a security zone to which interfaces can be assigned and enters security zone configuration mode.
Step 4	**description** *line-of-description* **Example:** `Router(config-sec-zone)# description Internet Traffic`	(Optional) Describes the zone.
Step 5	**exit** **Example:** `Router(config-sec-zone)# exit`	Returns to global configuration mode.
Step 6	**zone-pair security** *zone-pair-name* [**source** {*source-zone-name* | **self** | **default**} **destination** [*destination-zone-name* | **self** | **default**]] **Example:** `Router(config)# zone-pair security self-default-zp source self destination zone1`	Creates a zone pair and enters security zone-pair configuration mode. **Note** To apply a policy, you must configure a zone pair.
Step 7	**description** *line-of-description* **Example:** `Router(config-sec-zone-pair)# description accounting network`	(Optional) Describes the zone pair.
Step 8	**service-policy type inspect** *policy-map-name* **Example:** `Router(config-sec-zone-pair)# service-policy type inspect p1`	Attaches a firewall policy map to the destination zone pair. **Note** If a policy is not configured between a pair of zones, traffic is dropped by default.

	Command or Action	Purpose
Step 9	**exit** **Example:** `Router(config-sec-zone-pair)# exit`	Returns to global configuration mode.
Step 10	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet 0`	Configures an interface and enters interface configuration mode.
Step 11	**zone-member security** *zone-name* **Example:** `Router(config-if)# zone-member security zone1`	Assigns an interface to a specified security zone. **Note** When you make an interface a member of a security zone, all traffic into and out of that interface (except traffic bound for the router or initiated by the router) is dropped by default. To let traffic through the interface, you must make the zone part of a zone pair to which you apply a policy. If the policy permits traffic, traffic can flow through that interface.
Step 12	**end** **Example:** `Router(config-if)# end`	Exits interface configuration mode and enters privileged EXEC mode.

Configuring the Firewall with WAAS

Perform the following task to configure an end-to-end WAAS traffic flow optimization for the firewall that uses WCCP to redirect traffic to a WAE device for traffic interception.

In Cisco IOS XE software, WAAS support is always enabled and WAAS processing is always discovered. Thus, the **ip inspect waas enable** command is not necessary and therefore not supported.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip wccp** *service-id*
4. **class-map type inspect** [**match-any** | **match-all**] *class-map-name*
5. **match protocol** *protocol-name*
6. **exit**
7. **policy-map type inspect match-any** *policy-map-name*
8. **class type inspect** *class-name*
9. **inspect**
10. **class class-default**
11. **exit**
12. **exit**
13. **zone security** {*zone-name* | **default**}
14. **description** *line-of-description*
15. **exit**
16. **zone-pair security** *zone-pair-name* [**source** {*source-zone-name* | **self** | **default**} **destination** [*destination-zone-name*| **self** | **default**]]
17. **description** *line-of-description*
18. **service-policy type inspect** *policy-map-name*
19. **exit**
20. **interface** *type number*
21. **description** *line-of-description*
22. **zone-member security** *zone-name*
23. **ip address** *ip-address mask*
24. **ip wccp** {*service-id* {**group-listen** | **redirect** {**in** | **out**}} | **redirect exclude in** | **web-cache** {**group-listen** | **redirect** {**in** | **out**}}}
25. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip wccp** *service-id* **Example:** `Router(config)# ip wccp 61`	Enters the WCCP dynamically defined service identifier number.
Step 4	**class-map type inspect** [**match-any** | **match-all**] *class-map-name* **Example:** `Router(config)# class-map type inspect match-any most-traffic`	Creates an inspect type class map for the traffic class and enters QoS class-map configuration mode.
Step 5	**match protocol** *protocol-name* **Example:** `Router(config-cmap)# match protocol http`	Configures the match criteria for a class map based on the specified protocol. • Only Cisco IOS XE stateful packet inspection supported protocols can be used as match criteria in inspect type class maps. • **signature**—Signature-based classification for peer-to-peer (P2P) packets is enabled.
Step 6	**exit** **Example:** `Router(config-cmap)# exit`	Exits QoS class-map configuration mode and enters global configuration mode.
Step 7	**policy-map type inspect match-any** *policy-map-name* **Example:** `Router(config)# policy-map type inspect match-any p1`	Creates a Layer 3 or Layer 4 inspect type policy map and enters policy map configuration mode.

	Command or Action	Purpose
Step 8	**class type inspect** *class-name* **Example:** `Router(config-pmap)# class type inspect most-traffic`	Specifies the firewall traffic (class) map on which an action is to be performed. • Enters policy-map class configuration mode.
Step 9	**inspect** **Example:** `Router(config-pmap-c)# inspect`	Enables Cisco IOS XE stateful packet inspection.
Step 10	**class class-default** **Example:** `Router(config-pmap-c)# class class-default`	Specifies the matching of the system default class. • If the system default class is not specified, then unclassified packets are matched.
Step 11	**exit** **Example:** `Router(config-pmap-c)# exit`	Exits policy-map class configuration mode and enters policy map configuration mode.
Step 12	**exit** **Example:** `Router(config-pmap)# exit`	Exits policy map configuration mode and enters global configuration mode.
Step 13	**zone security** {*zone-name* | **default**} **Example:** `Router(config)# zone security zone1`	Creates a security zone to which interfaces can be assigned and enters security zone configuration mode.
Step 14	**description** *line-of-description* **Example:** `Router(config-sec-zone)# description Internet Traffic`	(Optional) Describes the zone.

	Command or Action	Purpose
Step 15	**exit** **Example:** `Router(config-sec-zone)# exit`	Returns to global configuration mode.
Step 16	**zone-pair security** *zone-pair-name* [**source** {*source-zone-name* | **self** | **default**} **destination** [*destination-zone-name*| **self** | **default**]] **Example:** `Router(config)# zone-pair security self-default-zp source self destination zone1`	Creates a zone pair and enters security zone-pair configuration mode. **Note** To apply a policy, you must configure a zone pair.
Step 17	**description** *line-of-description* **Example:** `Router(config-sec-zone-pair)# description accounting network`	(Optional) Describes the zone pair.
Step 18	**service-policy type inspect** *policy-map-name* **Example:** `Router(config-sec-zone-pair)# service-policy type inspect p1`	Attaches a firewall policy map to the destination zone pair. **Note** If a policy is not configured between a pair of zones, traffic is dropped by default.
Step 19	**exit** **Example:** `Router(config-sec-zone-pair)# exit`	Exits security zone-pair configuration mode and enters global configuration mode.
Step 20	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet 0/0/0`	Configures an interface and enters interface configuration mode.
Step 21	**description** *line-of-description* **Example:** `Router(config-if)# description 123`	(Optional) Describes the interface.

	Command or Action	Purpose
Step 22	**zone-member security** *zone-name* **Example:** `Router(config-if)# zone-member security zone1`	Assigns an interface to a specified security zone. **Note** When you make an interface a member of a security zone, all traffic into and out of that interface (except traffic bound for the router or initiated by the router) is dropped by default. To let traffic through the interface, you must make the zone part of a zone pair to which you apply a policy. If the policy permits traffic, traffic can flow through that interface.
Step 23	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 10.70.0.1 255.255.255.0`	Assigns the interface IP address for the security zone.
Step 24	**ip wccp** {*service-id* {**group-listen** | **redirect** {**in** | **out**}} | **redirect exclude in** | **web-cache** {**group-listen** | **redirect** {**in** | **out**}}} **Example:** `Router(config-if)# ip wccp 61 redirect in`	Specifies the following WCCP parameters on the interface: • The *service-id* argument defines a service identifier number from 1 to 254. • The **redirect exclude in** keywords are used to exclude inbound packets from outbound redirection. • The **web-cache** keyword is used to define the standard web caching service. • The **group-listen** keyword is used for discovering multicast WCCP protocol packets. • The **in** keyword is used to redirect the appropriate inbound packets to a cache engine. • The **out** keyword is used to redirect the appropriate outbound packets to a cache engine.
Step 25	**end** **Example:** `Router(config-if)# end`	Exits interface configuration mode and enters privileged EXEC mode.

Configuring an LDAP-Enabled Firewall

Lightweight Directory Access Protocol (LDAP) is an application protocol that is used for querying and updating information stored on directory servers. The LDAP-Enabled Firewall feature enables Cisco firewalls to support Layer 4 LDAP inspection by default.

You can configure an LDAP-enabled firewall in interface configuration mode or in global configuration mode. Before you configure an LDAP-enabled firewall in interface configuration mode, you must configure a zone by using the **zone security** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **zone security** {*zone-name* | **default**}
4. **exit**
5. **zone security** {*zone-name* | **default**}
6. **exit**
7. **class-map type inspect** [**match-any** | **match-all**] *class-map-name*
8. **match protocol** *protocol-name*
9. **exit**
10. **policy-map type inspect match-any** *policy-map-name*
11. **class type inspect** *class-name*
12. **inspect**
13. **class class-default**
14. **exit**
15. **exit**
16. **zone-pair security** *zone-pair-name* [**source** {*source-zone-name* | **self** | **default**} **destination** [*destination-zone-name* | **self** | **default**]]
17. **service-policy type inspect** *policy-map-name*
18. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**zone security** {*zone-name*	**default**} **Example:** `Router(config)# zone security private`	Creates a security zone to which interfaces can be assigned and enters security zone configuration mode.

	Command or Action	Purpose
Step 4	**exit** **Example:** `Router(config-sec-zone)# exit`	Exits security zone configuration mode and enters global configuration mode.
Step 5	**zone security** {*zone-name* | **default**} **Example:** `Router(config)# zone security internet`	Creates a security zone to which interfaces can be assigned and enters security zone configuration mode.
Step 6	**exit** **Example:** `Router(config-sec-zone)# exit`	Exits security zone configuration mode and enters global configuration mode.
Step 7	**class-map type inspect** [**match-any** | **match-all**] *class-map-name* **Example:** `Router(config)# class-map type inspect match-any internet-traffic-class`	Creates an inspect type class map for the traffic class and enters QoS class-map configuration mode.
Step 8	**match protocol** *protocol-name* **Example:** `Router(config-cmap)# match protocol ldap`	Configures the match criteria for a class map based on the specified protocol.
Step 9	**exit** **Example:** `Router(config-cmap)# exit`	Exits QoS class-map configuration mode and enters global configuration mode.
Step 10	**policy-map type inspect match-any** *policy-map-name* **Example:** `Router(config)# policy-map type inspect private-internet-policy`	Creates a Layer 3 or Layer 4 inspect type policy map and enters policy map configuration mode.

	Command or Action	Purpose
Step 11	**class type inspect** *class-name* **Example:** `Router(config-pmap)# class type inspect internet-traffic-class`	Specifies the firewall traffic (class) map on which an action is to be performed. • Enters policy-map class configuration mode.
Step 12	**inspect** **Example:** `Router(config-pmap-c)# inspect`	Enables Cisco IOS XE stateful packet inspection.
Step 13	**class class-default** **Example:** `Router(config-pmap-c)# class class-default`	Specifies the matching of the system default class. • If the system default class is not specified, then unclassified packets are matched.
Step 14	**exit** **Example:** `Router(config-pmap-c)# exit`	Exits policy-map class configuration mode and enters policy map configuration mode.
Step 15	**exit** **Example:** `Router(config-pmap)# exit`	Exits policy map configuration mode and enters global configuration mode.
Step 16	**zone-pair security** *zone-pair-name* [**source** {*source-zone-name* | **self** | **default**} **destination** [*destination-zone-name* | **self** | **default**]] **Example:** `Router(config)# zone-pair security private-internet source private destination internet`	Creates a zone pair and enters security zone-pair configuration mode. **Note** To apply a policy, you must configure a zone pair.
Step 17	**service-policy type inspect** *policy-map-name* **Example:** `Router(config-sec-zone-pair)# service-policy type inspect private-internet-policy`	Attaches a firewall policy map to the destination zone pair. **Note** If a policy is not configured between a pair of zones, traffic is dropped by default.

	Command or Action	Purpose
Step 18	**end** **Example:** `Router(config-sec-zone-pair)# end`	Exits security zone-pair configuration mode and enters privileged EXEC mode.

Configuration Examples for Zone-Based Policy Firewall

Example: Configuring Layer 3 or Layer 4 Firewall Policies

```
class-map type inspect match-all c1
 match access-group 101
 match protocol ftp
!
policy-map type inspect p1
 class type inspect c1
  inspect inspect-params
  pass
!
```

Example: Configuring an Inspect Parameter Map

```
parameter-map type inspect eng-network-profile
 alert on
 audit-trail on
 dns-timeout 60
 icmp idle-timeout 90
 max-incomplete low 800
 one-minute low 300
 sessions maximum 200
 tcp finwait-time 5
 tcp idle-time 90
 tcp max-incomplete host 500 block-time 10
 tcp synwait-time 3
 udp idle-time 75
```

Example: Creating Security Zones and a Zone Pair and Attaching a Policy Map to a Zone Pair

```
zone security zone1
 description Internet Traffic
!
```

```
zone-pair security self-default-zp source self destination zone1
 description accounting network
 service-policy type inspect p1
!
interface gigabitethernet 0
 zone-member security zone1
```

Example: Configuring NetFlow Event Logging

```
parameter-map type inspect global
 log dropped-packets
 log flow-export v9 udp destination 192.0.2.0 5000
 log flow-export template timeout rate 5000
```

Example: Firewall Configuration with WAAS

The following example provides an end-to-end WAAS traffic flow optimization configuration for the firewall that uses WCCP to redirect traffic to a WAE device for traffic interception.

The following configuration example prevents traffic from being dropped between security zone members because the integrated-service-engine interface is configured on a different zone and each security zone member is assigned an interface.

```
ip wccp 61
ip wccp 62
class-map type inspect match-any most-traffic
 match protocol icmp
 match protocol ftp
 match protocol tcp
 match protocol udp
policy-map type inspect p1
 class type inspect most--traffic
  inspect
  class class-default
zone security zone-hr
zone security zone-outside
zone security z-waas
zone-pair security hr-out source zone-hr destination zone-outside
 service-policy type inspect p1
zone-pair security out--hr source zone-outside destination zone-hr
 service-policy type inspect p1
zone-pair security eng-out source zone-eng destination zone-outside
 service-policy type inspect p1
interface GigabitEthernet 0/0/0
 description Trusted Interface
 ipaddress 10.70.0.1 255.0.0.0
 ip wccp 61 redirect in
 zone-member security zone-hr
interface GigabitEthernet 0/0/1
 description Trusted Interface
 ipaddress 10.71.0.2 255.0.0.0
 ip wccp 61 redirect in
 zone-member security zone-eng
interface GigabitEthernet 0/0/1
 description Untrusted Interface
 ipaddress 10.72.2.3 255.0.0.0
 ip wccp 62 redirect in
 zone-member security zone-outside
interface Integrated-Service-Engine l/0
 ipaddress 10.70.100.1 255.0.0.0
 ip wccp redirect exclude in
 zone-member security z-waas
```

Example: LDAP-Enabled Firewall Configuration

Interface Configuration

```
interface GigabitEthernet 0/1/5
ip address 192.168.0.1 255.255.255.0
zone-member security private
no shutdown
interface GigabitEthernet 0/1/6
ip address 192.168.1.1 255.255.255.0
zone-member security internet
no shutdown
```

Global Firewall Configuration

```
zone security private
zone security internet
class-map type inspect match-any internet-traffic-class
 match protocol ldap
 match protocol ldaps
 match protocol ldap-admin
policy-map type inspect private-internet-policy
 class type inspect internet-traffic-class
  inspect
  class class-default
zone-pair security private-internet source private destination internet
 service-policy type inspect private-internet-policy
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
Security commands	• *Cisco IOS Security Command Reference: Commands A to C* • *Cisco IOS Security Command Reference: Commands D to L* • *Cisco IOS Security Command Reference: Commands M to R* • *Cisco IOS Security Command Reference: Commands S to Z*
Quality of Service commands	*Cisco IOS Quality of Service Solutions Command Reference*
Per subscriber firewall support	*Release Notes for Cisco ASR 1000 Series Aggregation Services Routers for Cisco IOS XE Release 2*

Standards and RFCs

Standard/RFC	Title
RFC 4511	*Lightweight Directory Access Protocol (LDAP): The Protocol*

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Zone-Based Policy Firewall

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn . An account on Cisco.com is not required.

Table 1 Feature Information for Zone-Based Policy Firewall

Feature Name	Releases	Feature Configuration Information
Firewall High Speed Logging (HSL) Support	Cisco IOS XE Release 2.1	The Firewall High Speed Logging Support feature introduces support for the firewall HSL using NetFlow v9 as the export format. The following commands were introduced or modified: **log dropped-packet**, **log flow-export v9 udp destination**, **log flow-export template timeout-rate**, **parameter-map type inspect global**.
Firewall—NetMeeting Directory (LDAP) ALG Support	Cisco IOS XE Release 2.4	LDAP is an application protocol that is used for querying and updating information stored on directory servers. The Firewall—Netmeeting Directory ALG Support feature enables Cisco firewalls to support Layer 4 LDAP inspection by default. The following commands were introduced or modified: **match protocol**.
Out-of-Order Packet Handling in Zone-Based Policy Firewall	Cisco IOS XE Release 3.5S	The Out-of-Order Packet Handling feature allows OoO packets to pass through the router and reach their destination if a session does not require DPI. All Layer 4 traffic with OoO packets are allowed to pass through to their destination. However, if a session requires Layer 7 inspection, the OoO packets are still dropped.
Zone-Based Policy Firewall	Cisco IOS XE Release 2.1	The Zone-Based Policy Firewall feature provides a Cisco IOS XE software unidirectional firewall policy between groups of interfaces known as zones.

Feature Name	Releases	Feature Configuration Information
Zone-Based Firewall—Default Zone	Cisco IOS XE Release 2.6	The Zone-Based Firewall—Default Zone feature introduces a default zone that enables a firewall policy to be configured on a zone pair that consist of a zone and a default zone. Any interface without explicit zone membership belongs to a default zone. The following commands were introduced or modified: **zone-pair security** and **zone security**.

VRF-Aware Cisco IOS XE Firewall

The VRF-Aware Cisco IOS XE Firewall applies the Cisco IOS XE Firewall functionality to VPN Routing and Forwarding (VRF) interfaces when the firewall is configured on a service provider (SP) or large enterprise edge routers. SPs provide managed services to small and medium business markets.

The VRF-Aware Cisco IOS XE Firewall supports VRF-lite (also known as Multi-VRF CE) and Application Inspection and Control (AIC) for various protocols.

Note Cisco IOS XE Releases do not support Context-Based Access Control (CBAC) firewalls.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for VRF-Aware Cisco IOS XE Firewall

- Understand Cisco IOS XE firewalls.
- Configure VRFs.

Restrictions for VRF-Aware Cisco IOS XE Firewall

- If two VPN networks have overlapping addresses, VRF-aware Network Address Translation (NAT) is required for them to support VRF-aware firewalls. NAT does not support inter-VRF routing. You can use the VRF-aware software infrastructure (VASI) for the inter-VRF routing functionality.
- When crypto tunnels that belong to multiple VPNs terminate on a single interface, you cannot apply per-VRF firewall policies.
- The same zone cannot be applied to interfaces that are configured on different VRFs.

Information About VRF-Aware Cisco IOS XE Firewall

VRF-Aware Cisco IOS XE Firewall

A VRF-aware firewall inspects IP packets that are sent or received within a VRF. VRF allows multiple instances of routing tables to coexist within a single router. This allows VPN segregation and the ability to have independent overlapping of IP address spaces. VRF allows traffic from the customers of one service provider to be isolated from another. The Cisco IOS XE VRF support splits the router into multiple routing domains, with each routing domain consisting of its own set of interfaces and routing and forwarding tables. Each routing domain is referenced by a unique identifier called the table ID. The global routing domain and the default routing domain (that is not associated with any VRF) is addressed with the table ID, zero. VRF supports overlapping of IP address space, thereby allowing the traffic from nonintersecting VRFs to have the same IP address.

The VRF-Aware Cisco IOS XE Firewall provides the following benefits:

- Scalable deployment—Scales to meet any network's bandwidth and performance requirements.
- VPN support—Provides a complete VPN solution based on Cisco IOS XE IPsec and other software-based technologies, including Layer 2 Tunneling Protocol (L2TP) tunneling, and quality of service (QoS).
- AIC support—Provides policy maps for the Internet Message Access Protocol (IMAP), Post Office Protocol 3 (POP3), Simple Mail Transfer Protocol (SMTP), and Sun Remote Procedure Call (SUN RPC)

- Allows users to configure a per-VRF firewall. The firewall inspects IP packets that are sent and received within a VRF. The firewall also inspects traffic between two different VRFs (intersecting VRFs).
- Allows SPs to deploy the firewall on the provider edge (PE) router.
- Supports overlapping IP address space, thereby allowing traffic from nonintersecting VRFs to have the same IP address.
- Supports VRF (not global) firewall command parameters and Denial-of-Service (DoS) parameters so that the VRF-aware firewall can run as multiple instances (with VRF instances) that are allocated to various VPN customers.
- Generates high-speed logging (HSL) messages that contain the VRF ID; however these messages are collected by a single collector.

The VRF-aware firewall allows you to limit the number of firewall sessions. If the firewall sessions are not limited, it would be difficult for VRFs to share router resources because one VRF may consume a maximum amount of resources, leaving few resources for other VRFs and thereby causing the denial of service to other VRFs.

Address Space Overlap

A VRF splits the router into multiple routing domains. Each of these routing domains contain their own set of interfaces and routing tables. A routing table is referenced by using a per-VRF unique table ID. Zero is the default global routing table ID that is not associated with a VRF.

Nonintersecting VRFs are allowed to have overlapping address spaces (that is, the IP address of one VRF may be contained in others).

VRF

VRF allows multiple instances of routing tables to coexit within a single router. A VRF contains a template of a VRF table in a PE router.

The overlapping addresses, usually resulting from the use of private IP addresses in customer networks, are one of the major obstacles to the successful deployment of a peer-to-peer VPN implementation. You can use the MPLS VPN technology to overcome the overlapping addresses issue.

Each VPN has its own routing and forwarding table in the router so that any customer or site that belongs to a VPN is provided access only to the set of routes contained within that table. Any PE router in the MPLS VPN network therefore contains a number of per-VPN routing tables and a global routing table that is used to reach other routers in the service provider network. Effectively, a number of virtual routers are created in a single physical router.

VRF-Lite

The VRF-Lite Aware Firewall feature, also called the VRF without MPLS-aware firewall, allows a firewall zone to be applied to non-MPLS-enabled VRF interfaces.

The VRF-Lite Aware Firewall feature enables a service provider to support two or more VPNs, in which IP addresses can be overlapped among the VPNs. VRF-lite uses input interfaces to distinguish routes for different VPNs and forms virtual packet-forwarding tables by associating one or more Layer 3 interfaces with each VRF. Interfaces in a VRF can be physical, such as Ethernet ports, or logical, such as VLAN switched virtual interfaces (SVIs). However, a Layer 3 interface cannot belong to more than one VRF at a time.

Note All VRF-lite interfaces must be Layer 3 interfaces.

VRF-lite includes the following devices:

- Customer edge (CE) devices provide customers access to the service provider network over a data link. The CE device advertises the site's local routes to the PE router and learns about the remote VPN routes from the PE router.
- PE routers exchange routing information with CE devices by using static routing or a routing protocol such as Border Gateway Protocol (BGP), Routing Information Protocol Version 1 (RIPv1), or RIPv2.
- Provider routers (or core routers) are any routers in the service provider network that are not attached to CE devices.
- A PE router is only required to maintain VPN routes for those VPNs to which it is directly attached, eliminating the need for the PE router to maintain all the service provider VPN routes. Each PE router maintains a VRF for each of its directly connected sites. Multiple interfaces on a PE router can be associated with a single VRF, if all of these sites are part of the same VPN. Each VPN is mapped to a specified VRF. After learning local VPN routes from CE routers, a PE router exchanges VPN routing information with other PE routers by using internal BGP (iBPG).

With VRF-lite, multiple customers can share one CE router, and only one physical link is used between the CE router and the PE router. The shared CE router maintains a separate VRF table for each customer, and switches or routes packets for each customer based on its own routing table. VRF-lite extends the limited PE router functionality to a CE device, giving it the ability to maintain separate VRF tables to extend the privacy and security of a VPN to the branch office.

Figure 6 Firewall in a VRF-to-VRF Scenario

MPLS VPN

The MPLS VPN feature allows multiple sites to interconnect transparently through a service provider network. One service provider network can support several IP VPNs. Each VPN appears to its users as a private network, separate from all other networks. Within a VPN, each site can send IP packets to any other site in the same VPN.

Each VPN is associated with one or more VPN VRF instances. A VRF consists of an IP routing table, a derived Cisco Express Forwarding table, and a set of interfaces that use the forwarding table.

The router maintains a separate routing and Cisco Express Forwarding tables for each VRF. This prevents information from being sent outside the VPN and allows the same subnet to be used in several VPNs without causing duplicate IP address problems.

The router using Multiprotocol BGP (MP-BGP) distributes the VPN routing information using the MP-BGP extended communities.

VRF-Aware NAT

Network Address Translation (NAT) allows a single device, such as a router, to act as an agent between the Internet (or public network) and a local (or private) network. Although NAT systems can provide broad levels of security advantages, their main objective is to economize on address space.

NAT allows organizations to resolve the problem of IP address depletion when they have existing networks and need to access the Internet. Sites that do not possess Network Information Center (NIC)-registered IP addresses must acquire them. Cisco IOS XE NAT eliminates the concern of NIC-registered IP addresses by dynamically mapping thousands of hidden internal addresses to a range of easy-to-get addresses.

A NAT system makes it difficult for an attacker to determine the following:

- The number of systems running on a network.
- Type of machines and operating systems running on the network.
- Network topology and arrangement.

NAT integration with Multiprotocol Label Switching (MPLS) VPNs allows multiple MPLS VPNs to be configured on a single device to work together. NAT can differentiate the MPLS VPNs from which it receives the IP traffic, even if all the MPLS VPNs use the same IP addressing scheme. This enables multiple MPLS VPN customers to share services while ensuring that each MPLS VPN is completely separate from the other.

To provide value-added services, such as, Internet connectivity, domain name servers (DNS), and VoIP service to customers, the MPLS service providers must use NAT. NAT helps the MPLS VPN customers to use overlapped IP addresses in their network.

NAT can be implemented on a customer edge (CE) router or on a provider edge (PE) router. The NAT integration with the MPLS VPNs feature enables the implementation of NAT on a PE router in an MPLS cloud.

VRF-Aware ALG

An application-layer gateway (ALG) is an application that translates the IP address information inside the payload of an application packet. The ALGs identify the address information in the packet payload that needs to be overwritten by NAT and supply the address information to NAT and firewall to create subordinate flows or doors to allow data to flow properly (an example of data flow is FTP data flow. Doors are transient structures that allow incoming traffic that matches a specific criterion. A door is created when there is not enough information to create a complete NAT session entry. A door contains information about the source and destination IP address and the destination port. However, it does not have information about the source port. When media data arrives, the source port information is known and the door is promoted to a real NAT session.

VRF-Aware IPsec

The VRF-Aware IPsec feature maps an IPsec tunnel to an MPLS VPN. Using the VRF-Aware IPsec feature, you can map IPsec tunnels to VRF instances using a single public-facing IP address.

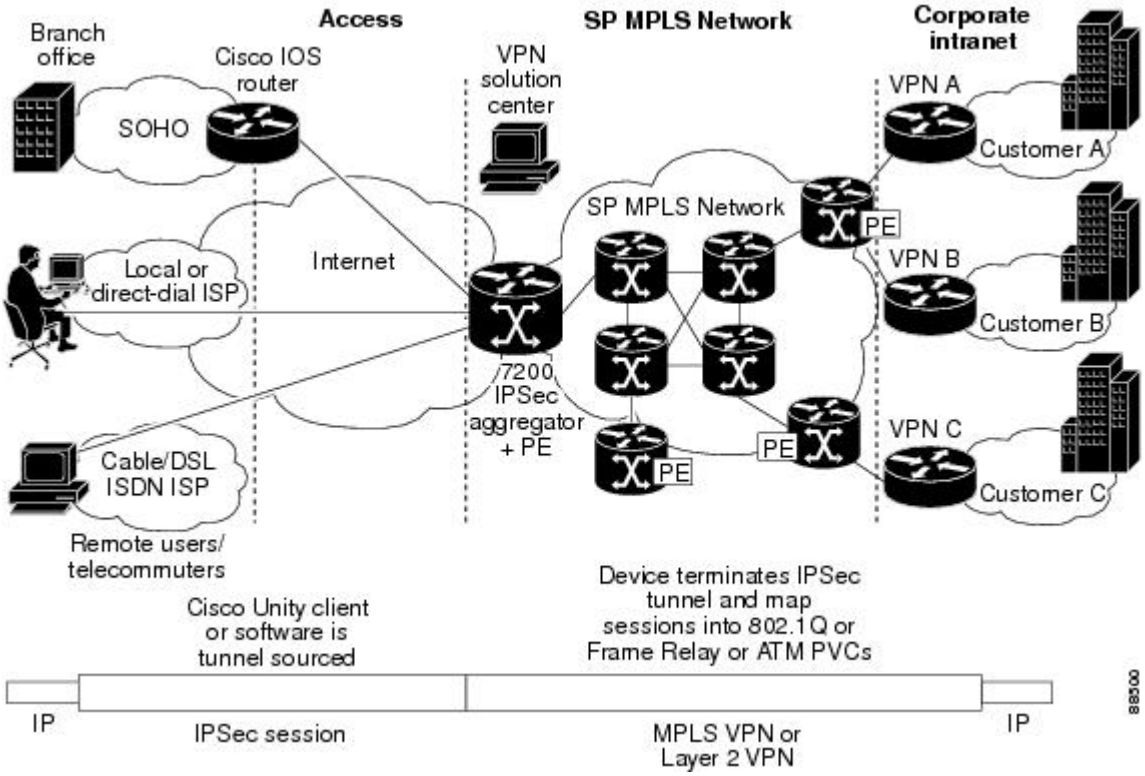
Each IPsec tunnel is associated with two VRF domains. The outer encapsulated packet belongs to a VRF domain called the Front Door VRF (FVRF). The inner, protected IP packet belongs to a domain called the Inside VRF (IVRF). In other words, the local endpoint of the IPsec tunnel belongs to the FVRF, whereas the source and destination addresses of the inside packet belong to the IVRF.

One or more IPsec tunnels can terminate on a single interface. The FVRF of all these tunnels is the same and is set to the VRF that is configured on that interface. The IVRF of these tunnels can be different and

depends on the VRF that is defined in the Internet Security Association and Key Management Protocol (ISAKMP) profile that is attached to a crypto map entry.

The figure below illustrates a scenario showing IPsec to MPLS and Layer 2 VPNs.

Figure 7 IPsec-to-MPLS and Layer 2 VPNs



VRF-Aware Software Infrastructure

The VRF-Aware Software Infrastructure (VASI) allows you to apply services such as access control lists (ACLs), NAT, policing, and zone-based firewalls to traffic that is flowing across two different VRF instances. The VASI interfaces support redundancy of the Route Processor (RP) and Forwarding Processor (FP). This feature supports IPv4 and IPv6 unicast traffic on VASI interfaces.

The primary use of VASI is to allow better isolation of VRFs. The VASI allows for per-VRF-specific features to be applied to the VASI interface without any impact to other VRFs that may share a common interface (for example, all VRFs may share the same interface to the Internet). For the firewall, this feature allows zones to be applied to the VASI.

VASI is implemented by using virtual interface pairs, where each of the interfaces in the pair is associated with a different VRF. The VASI virtual interface is the next hop interface for any packet that needs to be switched between these two VRFs. VASI interfaces provide the framework necessary to support NAT between two VRFs.

Each interface pair is associated with two different VRF instances. The two virtual interfaces, called vasileft and vasiright, in a pair are logically wired back-to-back and are completely symmetrical. Each interface has an index. The association of the pairing is done automatically based on the two interface indexes such that vasileft automatically gets paired to vasiright. You can configure either static routing or dynamic routing with BGP, Enhanced Interior Gateway Routing Protocol (EIGRP), or Open Shortest Path

First (OSPF). BGP dynamic routing protocol restrictions and configuration are valid for BGP routing configurations between VASI interfaces. For more information on VASI, see the "Configuring the VRF-Aware Software Infrastructure" feature.

Security Zones

A security zone is a group of interfaces to which a policy can be applied.

Grouping interfaces into zones involves two procedures:

- Creating a zone so that interfaces can be attached to it.
- Configuring an interface to be a member of a given zone.

By default, traffic flows among interfaces that are members of the same zone.

When an interface is a member of a security zone, all traffic (except traffic going to the router or initiated by the router) between that interface and an interface within a different zone is dropped by default. To permit traffic to and from a zone-member interface and another interface, you must make that zone part of a zone pair and apply a policy to that zone pair. If the policy permits traffic (through **inspect** or **pass** actions), traffic can flow through the interface.

Basic rules to consider when setting up zones are as follows:

- Traffic from a zone interface to a nonzone interface or from a nonzone interface to a zone interface is always dropped; unless default zones are enabled (default zone is a nonzone interface).
- Traffic between two zone interfaces is inspected if there is a zone-pair relationship for each zone and if there is a configured policy for that zone pair.
- By default, all traffic between two interfaces in the same zone is always allowed.
- A zone pair can be configured with a zone as both the source and the destination zones. An inspect policy can be configured on this zone pair to inspect or drop the traffic between two interfaces in the same zone.

A policy is applied to the initiating packet of a traffic flow. After the initial packet has been classified and permitted, traffic flows between peers with no further reclassification of the packet (this means that bidirectional traffic flow is allowed after the initial classification). If you have a zone pair between Zone Z1 and Zone Z2, and no zone pair between Zone Z2 and Zone Z1, all traffic that is initiated from Zone Z2 is blocked. Traffic from Zone Z1 to Zone Z2 is permitted or denied based on the zone pair policy.

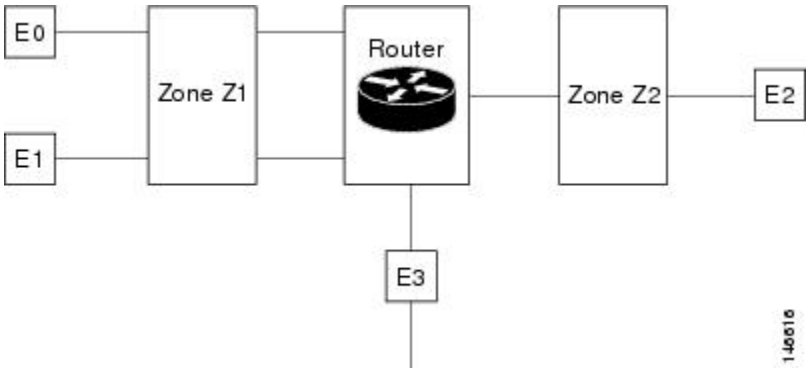
For traffic to flow among all the interfaces in a router, all interfaces must be members of security zones or the default zone.

It is not necessary for all router interfaces to be members of security zones.

The figure below illustrates the following:

- Interfaces E0 and E1 are members of security zone Z1.
- Interface E2 is a member of security zone Z2.
- Interface E3 is not a member of any security zone.

Figure 8 Security Zone Restrictions



The following situations exist:

- The zone pair and policy are configured in the same zone. If no policy is configured for Z1 and Z2, traffic will flow freely between E0 and E1, but not between E0 or E1 to E2. A zone pair and policy may be created to inspect this traffic.
- If no policies are configured, traffic will not flow between any other interfaces (for example, E0 and E2, E1 and E2, E3 and E1, and E3 and E2).
- Traffic can flow between E0 or E1 and E2 only when an explicit policy permitting traffic is configured between zone Z1 and zone Z2.
- Traffic can never flow between E3 and E0/E1/E2 unless default zones are enabled and a zone-pair is created between the default zone and the other zones.

VRF-Aware Cisco IOS XE Firewall Deployment

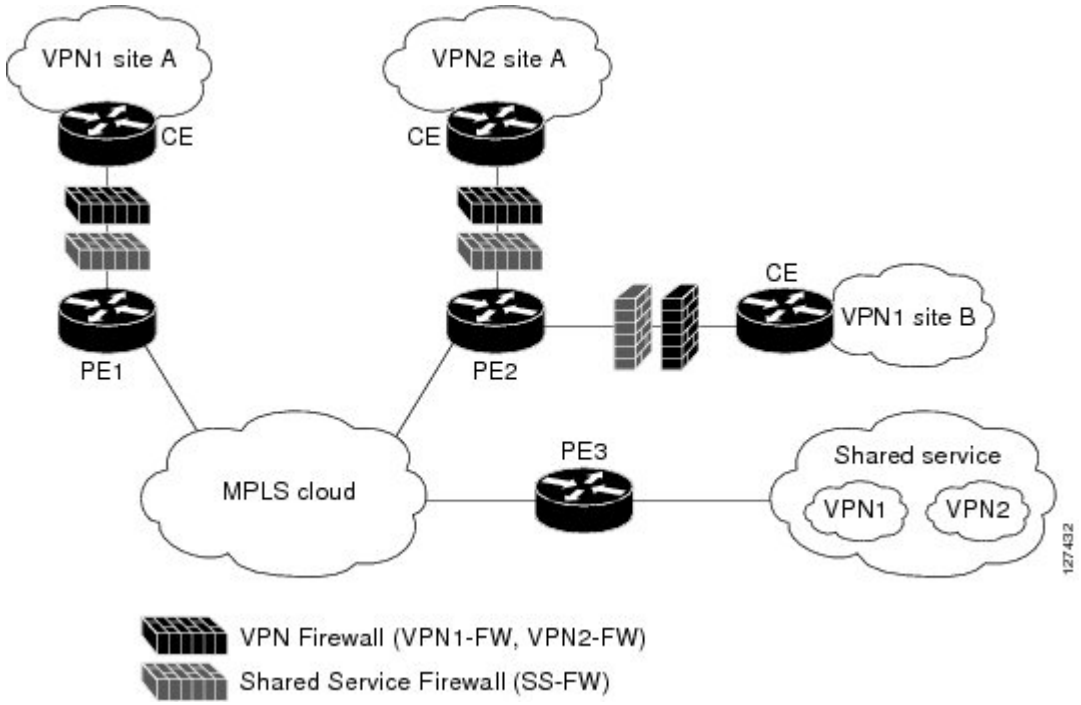
A firewall can be deployed at many points within the network to protect VPN sites from shared service (or the Internet) and vice versa. This section describes the following firewall deployment scenarios:

- Distributed Network Inclusion of VRF-Aware Cisco IOS XE Firewall, page 46
- Hub-and-Spoke Network Inclusion of VRF-Aware Cisco IOS XE Firewall, page 47

Distributed Network Inclusion of VRF-Aware Cisco IOS XE Firewall

The figure below illustrates a typical situation in which an SP offers firewall services to VPN customers VPN1 and VPN2, thereby protecting VPN sites from the external network (for example, shared services and the Internet) and vice versa.

Figure 9 Distributed Network



In this example, VPN1 has two sites, Site A and Site B, that span across the MPLS core. Site A is connected to PE1, and Site B is connected to PE2. VPN2 has only one site that is connected to PE2. Each VPN has a VLAN segment in the shared service that is connected to the corresponding VLAN subinterface on PE3.

Each of the VPNs (VPN1 and VPN2) has two firewall rules--one to protect the VPN site from the shared service and another to protect the shared service from the VPN site. The firewall that protects the VPN site from the shared service is called the VPN firewall, and the firewall that protects the shared service from the VPN site is called the Shared Service firewall. Both the firewall rules are applied on the VRF interface of each ingress PE that is connected to the VPN site. The VPN firewall rule is applied in the ingress direction, because the VRF interface is ingress to the VPN site; and the Shared Service Firewall rule is applied in the egress direction, because the VRF interface is egress to the shared service.

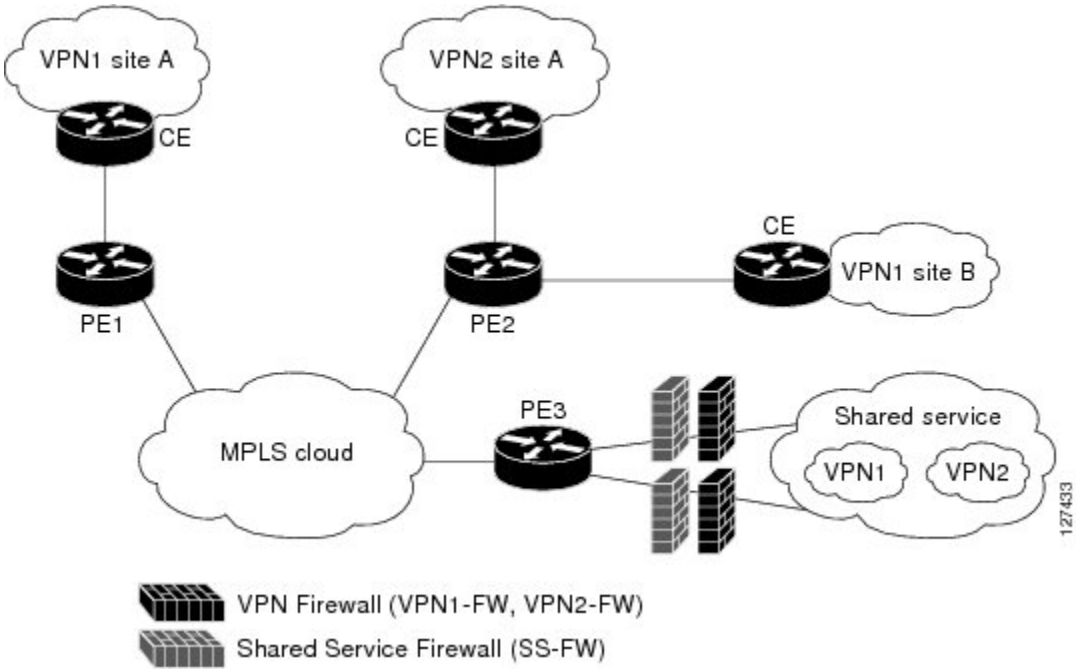
The benefits of using a distributed network are as follows:

- Because the firewall deployment is distributed across an MPLS cloud, the firewall processing load is distributed to all ingress PEs.
- The shared service is protected from VPN sites at the ingress PE, and hence malicious packets from VPN sites will be filtered at the ingress PE before they enter the MPLS cloud.
- VPN firewall features can be deployed in the ingress direction.

Hub-and-Spoke Network Inclusion of VRF-Aware Cisco IOS XE Firewall

The figure below illustrates a hub-and-spoke network where the firewalls for all VPN sites are applied on the egress PE router, PE3, which is connected to the shared service.

Figure 10 Hub-and-Spoke Network



Typically, each VPN has a VLAN and/or a VRF subinterface that is connected to the shared service. When a packet arrives at an MPLS interface, MPLS routes the packet to the corresponding subinterface that is connected to the shared service. The firewall policies on each VPN are applied on the corresponding

subinterface (VRF interface) as shown in the above figure. The VPN firewall rule is applied in the egress direction because the subinterface is egress to the VPN site. And the Shared Service firewall rule is applied in the ingress direction because the subinterface is ingress to the shared service.

The benefits of a hub-and-spoke network are as follows:

- Because the firewall deployment is centralized to the egress PE (PE3), deploying and managing the firewall is easy.
- The Shared Service firewall features can be applied in the ingress direction.
- The VPN site is protected from the shared service at the egress PE, and hence malicious packets from the shared service are filtered at the PE before they enter the MPLS cloud.

How to Configure VRF-Aware Cisco IOS XE Firewall

Defining VRFs, Class Maps, and Policy Maps

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip vrf** *vrf-name*
4. **rd** *route-distinguisher*
5. **route-target export** *route-target-ext-community*
6. **route-target import** *route-target-ext-community*
7. **exit**
8. **class-map type inspect match-any** *class-map-name*
9. **match protocol tcp**
10. **match protocol h323**
11. **exit**
12. **policy-map type inspect** *policy-map-name*
13. **class type inspect** *class-map-name*
14. **inspect** [*parameter-map-name*]
15. **exit**
16. **class class-default**
17. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip vrf** *vrf-name* **Example:** `Router(config)# ip vrf vrf1`	Defines a VRF instance and to enter VRF configuration mode.
Step 4	**rd** *route-distinguisher* **Example:** `Router(config-vrf)# rd 10:1`	Specifies a route distinguisher (RD) for a VRF instance.
Step 5	**route-target export** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target export 10:1`	Creates a route-target extended community for a VRF instance and exports routing information to the target VPN extended community.
Step 6	**route-target import** *route-target-ext-community* **Example:** `Router(config-vrf)# route-target import 10:1`	Creates a route-target extended community for a VRF instance and imports routing information to the target VPN extended community.
Step 7	**exit** **Example:** `Router(config-vrf)# exit`	Exits VRF configuration mode and enters global configuration mode.
Step 8	**class-map type inspect match-any** *class-map-name* **Example:** `Router(config)# class-map type inspect match-any class-map1`	Creates a Layer 3 and Layer 4 (application-specific) inspect type class map and enters class-map configuration mode.

	Command or Action	Purpose
Step 9	**match protocol tcp** **Example:** `Router(config-cmap)# match protocol tcp`	Configures the match criterion for a class map on the basis of the specified protocol.
Step 10	**match protocol h323** **Example:** `Router(config-cmap)# match protocol h323`	Configures the match criterion for a class map on the basis of the specified protocol.
Step 11	**exit** **Example:** `Router(config-cmap)# exit`	Exits class-map configuration mode and enters global configuration mode.
Step 12	**policy-map type inspect** *policy-map-name* **Example:** `Router(config)# policy-map type inspect global-vpn1-pmap`	Creates a Layer 3 and Layer 4 (protocol-specific) inspect type policy map and enters policy-map configuration mode.
Step 13	**class type inspect** *class-map-name* **Example:** `Router(config-pmap)# class type inspect class-map1`	Specifies the traffic (class) on which an action is to be performed and enters policy-map-class configuration mode.
Step 14	**inspect** [*parameter-map-name*] **Example:** `Router(config-pmap-c)# inspect class-map1`	Enables Cisco IOS XE stateful packet inspection.
Step 15	**exit** **Example:** `Router(config-pmap-c)# exit`	Exits policy-map-class configuration mode and enters policy-map configuration mode.
Step 16	**class class-default** **Example:** `Router(config-pmap)# class class-default`	Specifies the default class so that you can configure or modify its policy. • The class-default class is defined by default. Configure the **class class-default** command to change the default drop attribute that is associated with the class-default.

	Command or Action	Purpose
Step 17	**end** **Example:** `Router(config-pmap)# end`	Exits policy-map configuration mode and enters global configuration mode.

Defining Zones and Zone Pairs

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **zone security** *security-zone-name*
4. **exit**
5. **zone security** *security-zone-name*
6. **exit**
7. **zone-pair security** *zone-pair-name* **source** *source-zone* **destination** *destination-zone*
8. **service-policy type inspect** *policy-map-name*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**zone security** *security-zone-name* **Example:** `Router(config)# zone security vpn1-zone`	Creates a security zone and enters security zone configuration mode.

	Command or Action	Purpose
Step 4	**exit** **Example:** `Router(config-sec-zone)# exit`	Exits security zone configuration mode and enters global configuration mode.
Step 5	**zone security** *security-zone-name* **Example:** `Router(config)# zone security global-zone`	Creates a security zone and enters security zone configuration mode.
Step 6	**exit** **Example:** `Router(config-sec-zone)# exit`	Exits security zone configuration mode and enters global configuration mode.
Step 7	**zone-pair security** *zone-pair-name* **source** *source-zone* **destination** *destination-zone* **Example:** `Router(config)# zone-pair security vpn1-global-zone-pair source vpn1-zone destination global-zone`	Creates a zone pair and enters security zone-pair configuration mode. • *zone-pair-name*--Name of the zone being attached to an interface. • **source** *source-zone*--Specifies the name of the router from which traffic is originating. • **destination** *destination-zone*--Specifies the name of the router to which traffic is bound.
Step 8	**service-policy type inspect** *policy-map-name* **Example:** `Router(config-sec-zone-pair)# service-policy type inspect global-vpn1-pmap`	Attaches a Layer 7 policy map to a top-level policy map.
Step 9	**end** **Example:** `Router(config-sec-zone-pair)# end`	Exits zone-pair configuration mode and enters privileged EXEC mode.

Applying Zones to Interfaces and Defining Routes

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip vrf forwarding** *name*
5. **ip address** *ip-address mask*
6. **zone-member security** *zone-name*
7. **negotiation auto**
8. **exit**
9. **interface** *type number*
10. **ip address** *ip-address mask*
11. **zone-member security** *zone-name*
12. **negotiation auto**
13. **exit**
14. **ip route vrf** *vrf-name destination-ip-address destination-prefix interface-type number* [**global**]
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet 0/0/0`	Configures an interface and enters interface configuration mode.
Step 4	**ip vrf forwarding** *name* **Example:** `Router(config-if)# ip vrf forwarding vrf1`	Associates a VRF with an interface or subinterface.

	Command or Action	Purpose
Step 5	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 10.1.1.1 255.255.255.0`	Sets a primary or secondary IP address for an interface.
Step 6	**zone-member security** *zone-name* **Example:** `Router(config-if)# zone-member security  vpn1-zone`	Attaches an interface to a security zone.
Step 7	**negotiation auto** **Example:** `Router(config-if)# negotiation auto`	Enables the autonegotiation protocol to configure the speed, duplex, and automatic flow control of the Gigabit Ethernet interface.
Step 8	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 9	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet 1/1/1`	Configures an interface and enters interface configuration mode.
Step 10	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 10.111.111.111 255.255.255.0`	Sets a primary or secondary IP address for an interface.
Step 11	**zone-member security** *zone-name* **Example:** `Router(config-if)# zone-member security global-zone`	Attaches an interface to a security zone.
Step 12	**negotiation auto** **Example:** `Router(config-if)# negotiation auto`	Enables the autonegotiation protocol to configure the speed, duplex, and automatic flow control of the Gigabit Ethernet interface.

	Command or Action	Purpose
Step 13	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 14	**ip route vrf** *vrf-name destination-ip-address destination-prefix interface-type number* [**global**] **Example:** `Router(config)# ip route vrf vpn1 10.111.111.0 255.255.255.0 gigabitethernet 1/1/1 global`	Establishes static routes for a VRF instance.
Step 15	**end** **Example:** `Router(config)# end`	Exits global configuration mode and enters privileged EXEC mode.

Configuration Examples for VRF-Aware Cisco IOS XE Firewall

Example: Defining VRFs, Class Maps, and Policy Maps

```
Router# configure terminal
Router(config)# ip vrf vrf1
Router(config-vrf)# rd 10:1
Router(config-vrf)# route-target export 10:1
Router(config-vrf)# route-target import 10:1
Router(config-vrf)# exit
Router(config)# class-map type inspect match-any class-map1
Router(config-cmap)# match protocol tcp
Router(config-cmap)# match protocol h323
Router(config-cmap)# exit
Router(config)# policy-map type inspect global-vpn1-pmap
Router(config-pmap)# class type inspect match-acl-111
Router(config-pmap-c)# inspect match-acl-111
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap)# end
```

Example: Defining Policy Maps, Zones, and Zone Pairs

```
Router# configure terminal
Router(config)# zone security vpn1-zone
Router(config-sec-zone)# exit
Router(config)# zone security global-zone
Router(config-sec-zone)# exit
```

```
Router(config)# zone-pair security vpn1-global-zone-pair source vpn1-zone destination
global-zone
Router(config-sec-zone-pair)# service-policy type inspect vpn1-global-pmap
Router(config-sec-zone-pair)# end
```

Example: Applying Zones to Interfaces and Defining Routes

```
Router# configure terminal
Router(config)# interface gigabitethernet 0/0/0
Router(config-if)# ip vrf forwarding vrf1
Router(config-if)# ip address 10.1.1.1 255.255.255.0
Router(config-if)# zone-member security vpn1-zone
Router(config-if)# negotiation auto
Router(config-if)# exit
Router(config)# interface gigabitethernet 1/1/1
Router(config-if)# ip address 10.111.111.111 255.255.255.0
Router(config-if)# zone-member security global-zone
Router(config-if)# negotiation auto
Router(config-if)# exit
Router(config)# ip route vrf vpn1 10.111.111.0 255.255.255.0 gigabitethernet 1/1/1 global
Router(config)# end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
Security commands	• *Cisco IOS Security Command Reference Commands A to C* • *Cisco IOS Security Command Reference Commands D to L* • *Cisco IOS Security Command Reference Commands M to R* • *Cisco IOS Security Command Reference Commands S to Z*
NAT	*Configuring Network Address Translation: Getting Started*
MPLS VPN	*Configuring a Basic MPLS VPN*
Zone-based Policy Firewall	*Zone-based Policy Firewall*

Standards and RFCs

Standard/RFC	Title
No new or modified standards or RFCs are supported, and support for existing standards has not been modified.	--

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified.	To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for VRF-Aware Cisco IOS XE Firewall

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 2 Feature Information for VRF-Aware Cisco IOS XE Firewall

Feature Name	Releases	Feature Information
VRF-Aware Cisco IOS XE Firewall	Cisco IOS XE Release 2.5	The VRF-Aware Cisco IOS XE Firewall feature applies the Cisco IOS XE Firewall functionality to VRF interfaces when the firewall is configured on an SP or large enterprise edge router.

Feature Name	Releases	Feature Information
Firewall--VRF-Aware ALG Support	Cisco IOS XE Release 2.5	The Firewall--VRF-Aware ALG Support feature allows ALG to extract the correct IP address and VRF ID from cached information when creating ALG tokens that require correct IP address VRF ID pairs.

Glossary

CE router--customer edge router. A router that is part of a customer network and that interfaces to a provider edge router.

data authentication--Refers to one or both of the following: data integrity, which verifies that data has not been altered, or data origin authentication, which verifies that the data was actually sent by the claimed sender.

data confidentiality--A security service where the protected data cannot be observed.

edge router--A router that turns unlabeled packets into labeled packets, and vice versa.

firewall--A router or access server, or several routers or access servers, designated as a buffer between any connected public networks and a private network. A firewall router uses access lists and other methods to ensure the security of the private network.

IPsec--IP Security Protocol. A framework of open standards developed by the IETF. IPsec provides security for transmission of sensitive data over unprotected networks such as the Internet.

managed security services--A comprehensive set of programs that enhance service providers' abilities to meet the growing demands of their enterprise customers. Services based on Cisco solutions include managed firewall, managed VPN (network based and premises based), and managed intrusion detection.

NAT--Network Address Translation. Translates a private IP address used inside the corporation to a public, routable address for use outside of the corporation, such as the Internet. NAT is considered a one-to-one mapping of addresses from private to public.

PE router--provider edge router. A router that is part of a service provider's network and is connected to a customer edge (CE) router.

UDP--User Datagram Protocol. Connectionless transport layer protocol in the TCP/IP protocol stack. UDP is a simple protocol that exchanges datagrams without acknowledgments or guaranteed delivery, requiring that error processing and retransmission be handled by other protocols.

VPN--Virtual Private Network. Enables IP traffic to travel securely over a public TCP/IP network by encrypting all traffic from one network to another. A VPN uses "tunneling" to encrypt all information at the IP level.

vrf--A VPN routing and forwarding instance. A VRF consists of an IP routing table, a derived forwarding table, a set of interfaces that use the forwarding table, and a set of rules and routing protocols that determine what goes into the forwarding table. In general, a VRF includes the routing information that defines a customer VPN site that is attached to a provider edge (PE) router.

VRF table--A table that stores routing data for each VPN. The VRF table defines the VPN membership of a customer site attached to the network access server (NAS). Each VRF table comprises an IP routing table,

a derived Cisco Express Forwarding table, and guidelines and routing protocol parameters that control the information that is included in the routing table.

Cisco Firewall-SIP Enhancements ALG

The enhanced Session Initiation Protocol (SIP) inspection in the Cisco XE firewall provides basic SIP inspect functionality (SIP packet inspection and pinholes opening) as well as protocol conformance and application security. These enhancements give you control on what policies and security checks to apply to SIP traffic and the capability to filter out unwanted messages or users.

The development of additional SIP functionality in Cisco IOS XE software provides increased support for Cisco Call Manager, Cisco Call Manager Express, and Cisco IP-IP Gateway based voice/video systems. The application-layer gateway (ALG) SIP enhancement also supports RFC 3261 and its extensions.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Cisco Firewall-SIP Enhancements ALG

Your system must be running Cisco IOS XE Release 2.4 or a later release.

Restrictions for Cisco Firewall-SIP Enhancements ALG

DNS Name Resolution

Although SIP methods can have Domain Name System (DNS) names instead of raw IP addresses, this feature currently does not support DNS names.

Cisco ASR 1000 Series Routers

This feature was implemented without support for application inspection and control (AIC) on the Cisco ASR 1000 series routers. The Cisco IOS XE Release 2.4 supports the following commands only: **class-map type inspect**, **class type inspect**, **match protocol**, and **policy-map type inspect**.

Information About Cisco Firewall-SIP Enhancements ALG

SIP Overview

SIP is an application-layer control (signaling) protocol for creating, modifying, and terminating sessions with one or more participants. These sessions could include Internet telephone calls, multimedia distribution, and multimedia conferences. SIP is based on an HTTP-like request/response transaction model. Each transaction consists of a request that invokes a particular method or function on the server and at least one response.

SIP invitations that are used to create sessions carry session descriptions that allow participants to agree on a set of compatible media types. SIP makes use of elements called proxy servers to help route requests to the user's current location, authenticate and authorize users for services, implement provider call-routing policies, and provide features to users. SIP also provides a registration function that allows users to upload their current locations for use by proxy servers. SIP runs on top of several different transport protocols.

Firewall for SIP Functionality Description

The firewall for SIP support feature allows SIP signaling requests to traverse directly between gateways or through a series of proxies to the destination gateway or phone. After the initial request, if the Record-Route header field is not used, subsequent requests can traverse directly to the destination gateway address as specified in the Contact header field. Thus, the firewall is aware of all surrounding proxies and gateways and allows the following functionalities:

- SIP signaling responses can travel the same path as SIP signaling requests.
- Subsequent signaling requests can travel directly to the endpoint (destination gateway).
- Media endpoints can exchange data between each other.

SIP UDP and TCP Support

RFC 3261 is the current RFC for SIP, which replaces RFC 2543. This feature supports the SIP UDP and the TCP format for signaling.

SIP Inspection

This section describes the deployment scenarios supported by the Cisco Firewall--SIP ALG Enhancements feature.

Cisco IOS XE Firewall Between SIP Phones and CCM

The Cisco IOS XE firewall is located between Cisco Call Manager or Cisco Call Manager Express and SIP phones. SIP phones are registered to Cisco Call Manager or Cisco Call Manager Express through the firewall, and any SIP calls from or to the SIP phones pass through the firewall.

Cisco IOS XE Firewall Between SIP Gateways

The Cisco IOS XE firewall is located between two SIP gateways, which can be Cisco Call Manager, Cisco Call Manager Express, or a SIP proxy. Phones are registered with SIP gateways directly. The firewall sees the SIP session or traffic only when there is a SIP call between phones registered to different SIP gateways. In some scenarios an IP-IP gateway can also be configured on the same device as the firewall. With this scenario all the calls between the SIP gateways are terminated in the IP-IP gateway.

Cisco IOS XE Firewall with Local Cisco Call Manager Express and Remote Cisco Call Manager Express/ Cisco Call Manager

The Cisco IOS XE firewall is located between two SIP gateways, which can be Cisco Call Manager, Cisco Call Manager Express, or a SIP proxy. One of the gateways is configured on the same device as the firewall. All the phones registered to this gateway are locally inspected by the firewall. The firewall also inspects SIP sessions between the two gateways when there is a SIP call between them. With this scenario the firewall locally inspects SIP phones on one side and SIP gateways on the other side.

Cisco IOS XE Firewall with Local Cisco Call Manager Express

The Cisco IOS XE firewall and Cisco Call Manager Express is configured on the same device. All the phones registered to the Cisco Call Manager Express are locally inspected by the firewall. Any SIP call between any of the phones registered will also be inspected by the Cisco IOS XE firewall.

ALG--SIP Over TCP Enhancement

When SIP is transferred over UDP, every SIP message is carried in one single UDP datagram. However, when SIP is transferred over TCP, one TCP segment may contain multiple SIP messages. And it is possible that the last SIP message in one of the TCP segments may be a partial one. Prior to Cisco IOS XE Release 3.5S, when there are multiple SIP messages in one received TCP segment, the SIP ALG parses only the first message. The data that is not parsed is regarded as one incomplete SIP message and returned to vTCP. When the next TCP segment is received, vTCP prefixes the unprocessed data to that segment to pass them to the SIP ALG and causes more and more data have to be buffered in vTCP.

In Cisco IOS XE Release 3.5S, the ALG--SIP over TCP Enhancement feature lets the SIP ALG to handle multiple SIP messages in one TCP segment. When a TCP segment is received, all complete SIP messages inside this segment are parsed one-by-one. If there is an incomplete message in the end, only that portion is returned to vTCP.

How to Configure Cisco Firewall-SIP Enhancements ALG

Enabling SIP Inspection

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map type inspect match-any** *class-map-name*
4. **match protocol** *protocol-name*
5. **match protocol** *protocol-name*
6. **exit**
7. **policy-map type inspect** *policy-map-name*
8. **class type inspect** *class-map-name*
9. **inspect**
10. **exit**
11. **class class-default**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**class-map type inspect match-any** *class-map-name* **Example:** `Router(config)# class-map type inspect match-any sip-class1`	Creates an inspect type class map and enters class-map configuration mode.
Step 4	**match protocol** *protocol-name* **Example:** `Router(config-cmap)# match protocol sip`	Configures the match criterion for a class map based on the named protocol.

	Command or Action	Purpose
Step 5	**match protocol** *protocol-name* **Example:** `Router(config-cmap)# match protocol udp`	Configures the match criterion for a class map based on the named protocol.
Step 6	**exit** **Example:** `Router(config-cmap)# exit`	Exits class-map configuration mode.
Step 7	**policy-map type inspect** *policy-map-name* **Example:** `Router(config)# policy-map type inspect sip-policy`	Creates an inspect type policy map and enters policy map configuration mode.
Step 8	**class type inspect** *class-map-name* **Example:** `Router(config-pmap)# class type inspect sip-class1`	Specifies the class on which the action is performed and enters policy-map class configuration mode.
Step 9	**inspect** **Example:** `Router(config-pmap-c)# inspect`	Enables stateful packet inspection.
Step 10	**exit** **Example:** `Router(config-pmap-c)# exit`	Exits policy-map class configuration mode and enters policy map configuration mode.
Step 11	**class class-default** **Example:** `Router(config-pmap)# class class-default`	Specifies that these policy map settings apply to the predefined default class. If traffic does not match any of the match criteria in the configured class maps, it is directed to the predefined default class.
Step 12	**end** **Example:** `Router(config-pmap)# end`	Exits policy map configuration mode and enters privileged EXEC mode.

- Troubleshooting Tips, page 66

Troubleshooting Tips

The following commands can be used to troubleshoot your SIP-enabled firewall configuration:

- **clear zone-pair**
- **debug cce**
- **debug policy-map type inspect**
- **show policy-map type inspect zone-pair**
- **show zone-pair security**

Configuring a Zone Pair and Attaching a SIP Policy Map

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **zone security** {*zone-name* | **default**}
4. **exit**
5. **zone security** {*zone-name* | **default**}
6. **exit**
7. **zone-pair security** *zone-pair-name* [**source**{*source-zone-name* | **self** | **default**} **destination** [*destination-zone-name* | **self** | **default**]]
8. **service-policy type inspect** *policy-map-name*
9. **exit**
10. **interface** *type number*
11. **zone-member security** *zone-name*
12. **exit**
13. **interface** *type number*
14. **zone-member security** *zone-name*
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**zone security** {*zone-name* | **default**} **Example:** `Router(config)# zone security zone1`	Creates a security zone to which interfaces can be assigned and enters security zone configuration mode.
Step 4	**exit** **Example:** `Router(config-sec-zone)# exit`	Exits security zone configuration mode and enters global configuration mode.
Step 5	**zone security** {*zone-name* | **default**} **Example:** `Router(config)# zone security zone2`	Creates a security zone to which interfaces can be assigned and enters security zone configuration mode.
Step 6	**exit** **Example:** `Router(config-sec-zone)# exit`	Exits security zone configuration mode and enters global configuration mode.
Step 7	**zone-pair security** *zone-pair-name* [**source**{*source-zone-name* | **self** | **default**} **destination** [*destination-zone-name* | **self** | **default**]] **Example:** `Router(config)# zone-pair security in-out source zone1 destination zone2`	Creates a zone pair and enters security zone pair configuration mode. **Note** To apply a policy, you must configure a zone pair.
Step 8	**service-policy type inspect** *policy-map-name* **Example:** `Router(config-sec-zone-pair)# service-policy type inspect sip-policy`	Attaches a firewall policy map to the destination zone pair. **Note** If a policy is not configured between a pair of zones, traffic is dropped by default.
Step 9	**exit** **Example:** `Router(config-sec-zone-pair)# exit`	Exits security zone-pair configuration mode and enters global configuration mode.
Step 10	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet 0/0/0`	Configures an interface and enters interface configuration mode.

	Command or Action	Purpose
Step 11	**zone-member security** *zone-name* **Example:** `Router(config-if)# zone-member security zone1`	Assigns an interface to a specified security zone. **Note** When you make an interface a member of a security zone, all traffic in and out of that interface (except traffic bound for the router or initiated by the router) is dropped by default. To let traffic through the interface, you must make the zone part of a zone pair to which you apply a policy. If the policy permits traffic, traffic can flow through that interface.
Step 12	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 13	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet 0/1/1`	Configures an interface and enters interface configuration mode.
Step 14	**zone-member security** *zone-name* **Example:** `Router(config-if)# zone-member security zone2`	Assigns an interface to a specified security zone.
Step 15	**end** **Example:** `Router(config-if)# end`	Exits interface configuration mode and enters privileged EXEC mode.

Configuration Examples for Cisco Firewall-SIP Enhancements ALG

Example: Enabling SIP Inspection

```
class-map type inspect match-any sip-class1
 match protocol sip
 match protocol udp
!
```

```
policy-map type inspect sip-policy
    class type inspect sip-class1
        inspect
!
 class class-default
```

Example: Configuring a Zone-Pair and Attaching a SIP Policy Map

```
zone security zone1
!
zone security zone2
!
zone-pair security in-out source zone1 destination zone2
 service-policy type inspect sip-policy
!
interface gigabitethernet 0/0/0
 zone security zone1
!
interface gigabitethernet 0/1/1
 zone security zone2
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
Firewall commands	• *Cisco IOS Security Command Reference: Commands A to C* • *Cisco IOS Security Command Reference: Commands D to L* • *Cisco IOS Security Command Reference: Commands M to R* • *Cisco IOS Security Command Reference: Commands S to Z*
Additional SIP Information	*Guide to Cisco Systems VoIP Infrastructure Solution for SIP*
vTCP support	*vTCP for ALG Support*

Standards and RFCs

Standard/RFC	Title
RFC 3261	SIP: Session Initiation Protocol

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Cisco Firewall-SIP Enhancements ALG

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 3 Feature Information for Cisco Firewall-SIP Enhancements: ALG

Feature Name	Releases	Feature Information
AGL--SIP Over TCP Enhancement	Cisco IOS XE Release 3.5S	The ALG--SIP over TCP Enhancement feature lets the SIP ALG to handle multiple SIP messages in one TCP segment. When a TCP segment is received, all complete SIP messages inside this segment are parsed one-by-one. If there is an incomplete message in the end, only that portion is returned to vTCP.

Feature Name	Releases	Feature Information
Cisco Firewall--SIP ALG Enhancements	Cisco IOS XE Release 2.4	The Cisco Firewall--SIP ALG Enhancements feature provides voice security enhancements within the firewall feature set in Cisco IOS XE software on the Cisco ASR 1000 series routers. The following commands were implemented without support for Layer 7 (application-specific) syntax, on the Cisco ASR 1000 series routers:**class type inspect, class-map type inspect**, **match protocol**, **policy-map type inspect**.
Firewall--SIP ALG Enhancement for T.38 Fax Relay	Cisco IOS XE Release 2.4.1	The Firewall--SIP ALG Enhancement for T.38 Fax Relay feature provides an enhancement within the Firewall feature set in Cisco IOS XE software on the Cisco ASR 1000 series routers. The feature enables SIP ALG to support T.38 Fax Relay over IP, passing through the firewall on the Cisco ASR 1000 series routers.

Configuring the VRF-Aware Software Infrastructure Scale

This module describes how to configure the VRF-Aware Software Infrastructure Scale feature. The VRF-Aware Software Infrastructure (VASI) Scale feature allows you to apply services such as access control lists (ACLs), Network Address Translation (NAT), policing, and zone-based firewalls to traffic that is flowing across two different Virtual Routing and Forwarding (VRF) instances. The VASI interfaces support redundancy of the Route Processor (RP) and Forwarding Processor (FP). This feature supports Multiprotocol Label Switching (MPLS) traffic over VASI interfaces and IPv4 and IPv6 unicast traffic on VASI interfaces.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Configuring the VRF-Aware Software Infrastructure Scale

VASI interfaces do not support the attachment of queue-based features. The following commands are not supported on a modular quality of service (QoS) CLI (MQC) policy that is attached to VASI interfaces:

- **bandwidth (policy-map class)**
- **fair-queue**
- **priority**

- **queue-limit**
- **random-detect**
- **shape**

Information About Configuring the VRF-Aware Software Infrastructure Scale

VASI Overview

VASI is implemented by using virtual interface pairs, where each of the interfaces in the pair is associated with a different VRF. The VASI virtual interface is the next hop interface for any packet that needs to be switched between these two VRFs. VASI interfaces provide the framework necessary to configure a firewall or a NAT between VRF instances.

Each interface pair is associated with two different VRF instances. The two virtual interfaces, called vasileft and vasiright, in a pair are logically wired back-to-back and are completely symmetrical. Each interface has an index. The association of the pairing is done automatically based on the two interface indexes such that vasileft automatically gets paired to vasiright. You can configure either static routing or dynamic routing with Border Gateway Protocol (BGP), Enhanced Interior Gateway Routing Protocol (EIGRP), or Open Shortest Path First (OSPF). BGP dynamic routing protocol restrictions and configuration are valid for BGP routing configurations between VASI interfaces.

How to Configure VASI

Configuring the VASI Interface

VASI must be enabled on both interfaces of the VASI pair (vasileft and vasiright). You can configure VRF on any VASI interface. Perform the following task to configure the VASI interfaces.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface vasileft** *number*
4. **vrf forwarding** *table-name* [**downstream** *table-name*]
5. **ip address** {*ip-address mask* [**secondary**] | **pool pool-name**}
6. **exit**
7. **interface vasiright** *number*
8. **vrf forwarding** *table-name* [**downstream**
9. **ip address** {*ip-address mask* [**secondary**] | **pool pool-name**}
10. **exit**
11. **ip route** [**vrf** *vrf-name*] *destination-prefix destination-prefix-mask*{**vasileft** | **vasiright**} *number*
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface vasileft** *number* **Example:** `Router(config)# interface vasileft 200`	Configures the vasileft interface and enters interface configuration mode. • *number* --A number for the vasileft interface. Range is from 1 to 1000.
Step 4	**vrf forwarding** *table-name* [**downstream** *table-name*] **Example:** `Router(config-if)# vrf forwarding table1`	Configures the VRF table. **Note** You can configure VRF forwarding on any VASI interface. It is not mandatory to configure VRF instances on both VASI interfaces.

	Command or Action	Purpose
Step 5	**ip address** {*ip-address mask* [**secondary**] | **pool pool-name**} **Example:** `Router(config-if)# ip address 192.168.0.1 255.255.255.0`	Configures a primary or secondary IP address for an interface.
Step 6	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 7	**interface vasiright** *number* **Example:** `Router(config)# interface vasiright 200`	Configures the vasiright interface and enters interface configuration mode. • *number* --A number for the vasiright interface. Range is from 1 to 1000.
Step 8	**vrf forwarding** *table-name* [**downstream** **Example:** *table-name*] **Example:** `Router(config-if)# vrf forwarding table`	Configures the VRF table.
Step 9	**ip address** {*ip-address mask* [**secondary**] | **pool pool-name**} **Example:** `Router(config-if)# ip address 192.168.1.1 255.255.255.0`	Configures a primary or secondary IP address for an interface.
Step 10	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.

	Command or Action	Purpose
Step 11	**ip route** [**vrf** *vrf-name*] *destination-prefix destination-prefix-mask*{**vasileft** | **vasiright**} *number* **Example:** `Router(config)# ip route vrf t1 10.0.0.1 255.255.0.0 vasileft 200`	Establishes static routes for a VRF instance and VASI interface. **Note** If you want to add an IP route for a VRF instance, you must specify the **vrf** keyword.
Step 12	**end** **Example:** `Router(config)# end`	Exits global configuration mode.

Configuration Examples for VASI

Example Configuring the VASI Interface

The following example shows how to configure the VASI interface. VASI must be enabled for each interface of the VASI pair (vasileft and vasiright). You can configure VRF on any VASI interface. See the Configuring the VASI Interface section for configuration information.

```
Router(config)# interface vasileft 200
Router(config-if)# vrf forwarding table1

Router(config-if)# ip address 192.168.0.1 255.255.255.0

Router(config-if)# exit


Router(config)# ip route vrf t1 10.0.0.1 255.255.0.0 vasileft 200

Router(config)# interface vasiright 200

Router(config-if)# vrf forwarding table2

Router(config-if)# ip address 192.168.1.1 255.255.255.0

Router(config-if)# exit

Router(config)# ip route 10.0.0.2 255.255.255.0 vasiright 200
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Security commands	*Cisco IOS Security Command Reference*
Configuring NAT for IP Address Conservation feature	" Configuring NAT for IP Address Conservation" module of the *IP Addressing Configuration Guide*
IP routing: BGP	*IP Routing: BGP Configuration Guide, Cisco IOS XE Release*
IP routing: EIGRP	*IP Routing: EIGRP Configuration Guide, Cisco IOS XE Release*
IP routing: OSPF	*IP Routing: OSPF Configuration Guide, Cisco IOS XE Release*
VRF Aware Cisco IOS Firewall feature	"VRF Aware Cisco IOS Firewall"module of the *Security Configuration Guide: Securing the Control Plane*
Zone-based Policy Firewall feature	"Zone-based Policy Firewall"module of the *Security Configuration Guide: Securing the Control Plane*

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified.	--

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Configuring VRF-Aware Software Infrastructure Scale

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 4 Feature Information for Configuring the VRF-Aware Software Infrastructure Scale

Feature Name	Releases	Feature Information
Configuring VRF-Aware Software Infrastructure Scale	Cisco IOS XE Release 2.6	The VRF-Aware Software Infrastructure (VASI) Scale feature allows you to apply services such as ACLs, NAT, policing, and zone-based firewalls to traffic that is flowing across two different VRF instances. The VASI interfaces support redundancy of the RP and FP. This feature supports MPLS traffic over VASI interfaces and IPv4 and IPv6 multicast and unicast traffic on VASI interfaces. The following sections provide information about this feature:

Feature Name	Releases	Feature Information
VASI (VRF-Aware Software Infrastructure) Enhancements Phase I	Cisco IOS XE Release 3.1S	This feature provides the following enhancements to VASI: • Support for 500 VASI interfaces. • Support for BGP dynamic routing between VASI interfaces.
VASI (VRF-Aware Software Infrastructure) Enhancements Phase II	Cisco IOS XE Release 3.2S	This feature provides the following enhancements to VASI: • Support for IPv6 unicast traffic over VASI interfaces. • Support for OSPF and EIGRP dynamic routing between VASI interfaces.
VASI (VRF-Aware Software Infrastructure) Scale	Cisco IOS XE Release 3.3S	This feature provides support for 1000 VASI interfaces. The following commands were introduced or modified: **interface (VASI)**.