



IP Configuration Guide, Cisco IOS XE Gibraltar 16.10.x (Catalyst 9500 Switches)

First Published: 2018-11-15

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

CHAPTER 1

Configuring HSRP 1

Configuring HSRP 1

Finding Feature Information 1

Information About Configuring HSRP 1

HSRP Overview 1

HSRP Versions 3

Multiple HSRP 4

HSRP and Switch Stacks 4

Configuring HSRP for IPv6 4

How to Configure HSRP 5

Default HSRP Configuration 5

HSRP Configuration Guidelines 5

Enabling HSRP 6

Configuring HSRP Priority 7

Configuring MHSRP 10

Configuring HSRP Authentication and Timers 16

Enabling HSRP Support for ICMP Redirect Messages 17

Configuring HSRP Groups and Clustering 17

Verifying HSRP 18

Verifying HSRP Configurations 18

Configuration Examples for Configuring HSRP 18

Enabling HSRP: Example 18

Configuring HSRP Priority: Example 19

Configuring MHSRP: Example 19

Configuring HSRP Authentication and Timer: Example 19

Configuring HSRP Groups and Clustering: Example 20

Additional References for Configuring HSRP	20
Feature Information for Configuring HSRP	21

CHAPTER 2**Configuring NHRP 23**

Configuring NHRP	23
Information About Configuring NHRP	23
NHRP and NBMA Network Interaction	23
Dynamically Built Hub-and-Spoke Networks	24
How to Configure NHRP	24
Enabling NHRP on an Interface	24
Configuring a GRE Tunnel for Multipoint Operation	25
Configuration Examples for NHRP	28
Physical Network Designs for Logical NBMA Examples	28
Example: GRE Tunnel for Multipoint Operation	29
Additional References for Configuring NHRP	30
Feature Information for Configuring NHRP	30

CHAPTER 3**Configuring Network Address Translation 33**

Finding Feature Information	33
Network Address Translation (NAT)	34
Benefits of Configuring NAT	34
How NAT Works	34
Uses of NAT	35
NAT Inside and Outside Addresses	35
Types of NAT	36
Using NAT to Route Packets to the Outside Network (Inside Source Address Translation)	37
Outside Source Address Translation	38
Port Address Translation (PAT)	38
Overlapping Networks	40
Limitations of NAT	41
Performance and Scale Numbers for NAT	42
Address Only Translation	42
Restrictions for Address Only Translation	42
Configuring NAT	43

Configuring Static Translation of Inside Source Addresses	43
Configuring Dynamic Translation of Inside Source Addresses	45
Configuring PAT	47
Configuring NAT of External IP Addresses Only	49
Configuring Translation of Overlapping Networks	50
Configuring Address Translation Timeouts	52
Configuring Switch Database Management (SDM) Template	54
Using Application-Level Gateways with NAT	55
Best Practices for NAT Configuration	55
Troubleshooting NAT	56
Feature Information for Network Address Translation	56

CHAPTER 4**VRRPv3 Protocol Support 57**

VRRPv3 Protocol Support	57
Restrictions for VRRPv3 Protocol Support	57
Information About VRRPv3 Protocol Support	58
VRRPv3 Benefits	58
VRRP Device Priority and Preemption	59
VRRP Advertisements	60
How to Configure VRRPv3 Protocol Support	60
Creating and Customizing a VRRP Group	60
Configuring the Delay Period Before FHRP Client Initialization	62
Configuration Examples for VRRPv3 Protocol Support	63
Example: Enabling VRRPv3 on a Device	63
Example: Creating and Customizing a VRRP Group	63
Example: Configuring the Delay Period Before FHRP Client Initialization	64
Example: VRRP Status, Configuration, and Statistics Details	64
Additional References	65
Feature Information for VRRPv3 Protocol Support	66
Glossary	66

CHAPTER 5**Configuring WCCP 67**

Introduction	67
Prerequisites for WCCP	67

Restrictions for WCCP	67
Information About WCCP	69
WCCP Overview	69
WCCP Mask Assignment	69
WCCPv2 Configuration	69
WCCPv2 Support for Services Other Than HTTP	71
WCCPv2 Support for Multiple Devices	71
WCCPv2 MD5 Security	71
WCCPv2 Web Cache Packet Return	71
WCCPv2 Load Distribution	72
WCCP Bypass Packets	72
WCCP Closed Services and Open Services	72
WCCP Outbound ACL Check	72
WCCP Service Groups	73
WCCP—Check All Services	74
WCCP Troubleshooting Tips	74
How to Configure WCCP	74
Configuring WCCP	75
Configuring Closed Services	76
Registering a Device to a Multicast Address	78
Using Access Lists for a WCCP Service Group	79
Enabling the WCCP Outbound ACL Check	81
Verifying and Monitoring WCCP Configuration Settings	82
Configuration Examples for WCCP	83
Example: Configuring a General WCCPv2 Session	83
Example: Setting a Password for a Device and Content Engines	83
Example: Configuring a Web Cache Service	83
Example: Running a Reverse Proxy Service	84
Example: Registering a Device to a Multicast Address	84
Example: Using Access Lists	84
Example: WCCP Outbound ACL Check Configuration	85
Example: Verifying WCCP Settings	85
Feature Information for WCCP	87

CHAPTER 6	Configuring Enhanced Object Tracking	89
	Finding Feature Information	89
	Restrictions for Enhanced Object Tracking	89
	Information About Enhanced Object Tracking	90
	Enhanced Object Tracking Overview	90
	Tracking Interface Line-Protocol or IP Routing State	90
	Tracked Lists	90
	Tracking Other Characteristics	91
	IP SLAs Object Tracking	91
	Static Route Object Tracking	91
	How to Configure Enhanced Object Tracking	92
	Configuring Tracking for Line State Protocol or IP Routing State on an Interface	92
	Configuring Tracked Lists	93
	Configuring a Tracked List with a Weight Threshold	93
	Configuring a Tracked List with a Percentage Threshold	95
	Configuring HSRP Object Tracking	96
	Configuring IP SLAs Object Tracking	98
	Configuring Static Route Object Tracking	99
	Configuring a Primary Interface for Static Routing	99
	Configuring a Primary Interface for DHCP	100
	Configuring IP SLAs Monitoring Agent	101
	Configuring a Routing Policy and a Default Route	102
	Monitoring Enhanced Object Tracking	104
	Feature History for Enhanced Object Tracking	104
CHAPTER 7	Configuring TCP MSS Adjustment	107
	Restrictions for TCP MSS Adjustment	107
	Information about TCP MSS Adjustment	107
	Configuring the MSS Value for Transient TCP SYN Packets	108
	Configuring the MSS Value for IPv6 Traffic	109
	Example: Configuring the TCP MSS Adjustment	110
	Example: Configuring the TCP MSS Adjustment for IPv6 traffic	110
	Feature History for TCP MSS Adjustment	110

CHAPTER 8	Enhanced IPv6 Neighbor Discovery Cache Management	111
	Enhanced IPv6 Neighbor Discovery Cache Management	111
	Customizing the Parameters for IPv6 Neighbor Discovery	112
	Examples: Customizing Parameters for IPv6 Neighbor Discovery	113
	Additional References	113
	Feature Information for IPv6 Neighbor Discovery	113



CHAPTER 1

Configuring HSRP

- [Configuring HSRP](#) , on page 1

Configuring HSRP

This chapter describes how to use Hot Standby Router Protocol (HSRP) to provide routing redundancy for routing IP traffic without being dependent on the availability of any single router.

You can also use a version of HSRP in Layer 2 mode to configure a redundant command switch to take over cluster management if the cluster command switch fails. This feature is not supported on the C9500-12Q, C9500-16X, C9500-24Q, C9500-40X models of the Cisco Catalyst 9500 Series Switches.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfngn.cisco.com/>. An account on Cisco.com is not required.

Information About Configuring HSRP

HSRP Overview

HSRP is Cisco's standard method of providing high network availability by providing first-hop redundancy for IP hosts on an IEEE 802 LAN configured with a default gateway IP address. HSRP routes IP traffic without relying on the availability of any single router. It enables a set of router interfaces to work together to present the appearance of a single virtual router or default gateway to the hosts on a LAN. When HSRP is configured on a network or segment, it provides a virtual Media Access Control (MAC) address and an IP address that is shared among a group of configured routers. HSRP allows two or more HSRP-configured routers to use the MAC address and IP network address of a virtual router. The virtual router does not exist; it represents the common target for routers that are configured to provide backup to each other. One of the routers is selected to be the active router and another to be the standby router, which assumes control of the group MAC address and IP address should the designated active router fail.



Note Routers in an HSRP group can be any router interface that supports HSRP, including routed ports and switch virtual interfaces (SVIs).

HSRP provides high network availability by providing redundancy for IP traffic from hosts on networks. In a group of router interfaces, the active router is the router of choice for routing packets; the standby router is the router that takes over the routing duties when an active router fails or when preset conditions are met.

HSRP is useful for hosts that do not support a router discovery protocol and cannot switch to a new router when their selected router reloads or loses power. When HSRP is configured on a network segment, it provides a virtual MAC address and an IP address that is shared among router interfaces in a group of router interfaces running HSRP. The router selected by the protocol to be the active router receives and routes packets destined for the group's MAC address. For n routers running HSRP, there are $n + 1$ IP and MAC addresses assigned.

HSRP detects when the designated active router fails, and a selected standby router assumes control of the Hot Standby group's MAC and IP addresses. A new standby router is also selected at that time. Devices running HSRP send and receive multicast UDP-based hello packets to detect router failure and to designate active and standby routers. When HSRP is configured on an interface, Internet Control Message Protocol (ICMP) redirect messages are automatically enabled for the interface.

You can configure multiple Hot Standby groups among switches and switch stacks that are operating in Layer 3 to make more use of the redundant routers.

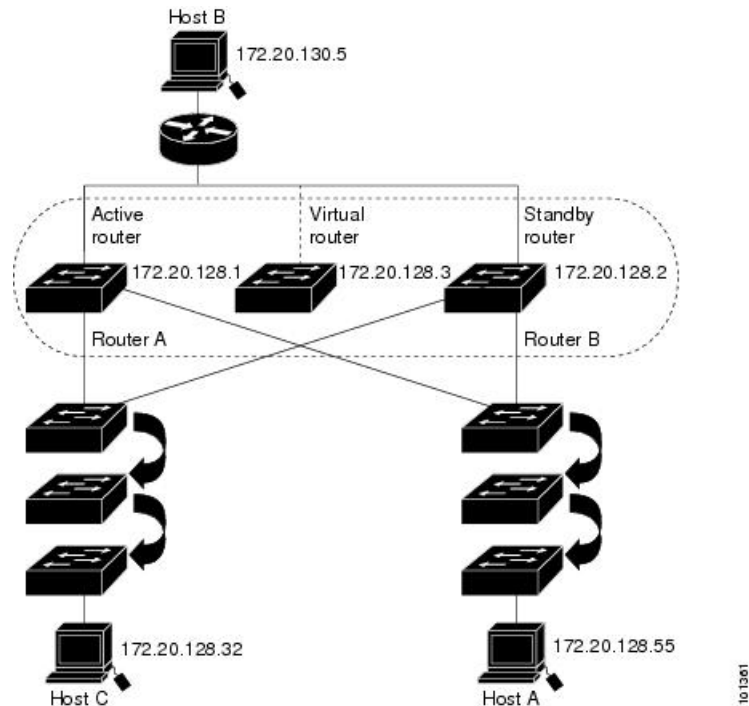


Note Catalyst 9500-H Series Switches do not support stacking.

To do so, specify a group number for each Hot Standby command group you configure for an interface. For example, you might configure an interface on switch 1 as an active router and one on switch 2 as a standby router and also configure another interface on switch 2 as an active router with another interface on switch 1 as its standby router.

The following figure shows a segment of a network configured for HSRP. Each router is configured with the MAC address and IP network address of the virtual router. Instead of configuring hosts on the network with the IP address of Router A, you configure them with the IP address of the virtual router as their default router. When Host C sends packets to Host B, it sends them to the MAC address of the virtual router. If for any reason, Router A stops transferring packets, Router B responds to the virtual IP address and virtual MAC address and becomes the active router, assuming the active router duties. Host C continues to use the IP address of the virtual router to address packets destined for Host B, which Router B now receives and sends to Host B. Until Router A resumes operation, HSRP allows Router B to provide uninterrupted service to users on Host C's segment that need to communicate with users on Host B's segment and also continues to perform its normal function of handling packets between the Host A segment and Host B.

Figure 1: Typical HSRP Configuration



HSRP Versions

Cisco IOS XE Everest 16.5.1a and later support these Hot Standby Router Protocol (HSRP) versions:

The switch supports these HSRP versions:

- HSRPv1- Version 1 of the HSRP, the default version of HSRP. It has these features:
 - The HSRP group number can be from 0 to 255.
 - HSRPv1 uses the multicast address 224.0.0.2 to send hello packets, which can conflict with Cisco Group Management Protocol (CGMP) leave processing. You cannot enable HSRPv1 and CGMP at the same time; they are mutually exclusive.
- HSRPv2- Version 2 of the HSRP has these features:
 - HSRPv2 uses the multicast address 224.0.0.102 to send hello packets. HSRPv2 and CGMP leave processing are no longer mutually exclusive, and both can be enabled at the same time.
 - HSRPv2 has a different packet format than HSRPv1.

A switch running HSRPv1 cannot identify the physical router that sent a hello packet because the source MAC address of the router is the virtual MAC address.

HSRPv2 has a different packet format than HSRPv1. A HSRPv2 packet uses the type-length-value (TLV) format and has a 6-byte identifier field with the MAC address of the physical router that sent the packet.

If an interface running HSRPv1 gets an HSRPv2 packet, the type field is ignored.

Multiple HSRP

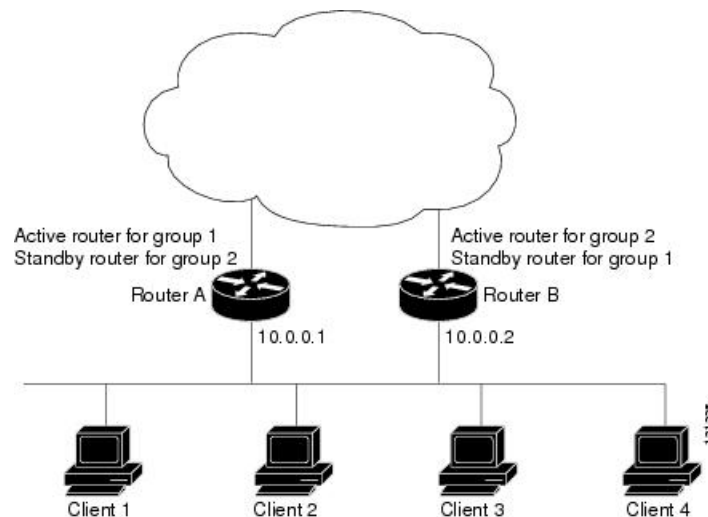
The switch supports Multiple HSRP (MHSRP), an extension of HSRP that allows load sharing between two or more HSRP groups. You can configure MHSRP to achieve load-balancing and to use two or more standby groups (and paths) from a host network to a server network.

In the figure below, half the clients are configured for Router A, and half the clients are configured for Router B. Together, the configuration for Routers A and B establishes two HSRP groups. For group 1, Router A is the default active router because it has the assigned highest priority, and Router B is the standby router. For group 2, Router B is the default active router because it has the assigned highest priority, and Router A is the standby router. During normal operation, the two routers share the IP traffic load. When either router becomes unavailable, the other router becomes active and assumes the packet-transfer functions of the router that is unavailable.



Note For MHSRP, you need to enter the **standby preempt** interface configuration command on the HSRP interfaces so that if a router fails and then comes back up, preemption restores load sharing.

Figure 2: MHSRP Load Sharing



HSRP and Switch Stacks

This feature is not supported on the C9500-12Q, C9500-16X, C9500-24Q, C9500-40X models of the Cisco Catalyst 9500 Series Switches. HSRP hello messages are generated by the active switch. If HSRP fails on the active switch, a flap in the HSRP active state might occur. This is because HSRP hello messages are not generated while a new active switch is elected and initialized, and the standby switch might become active after the active switch fails.

Configuring HSRP for IPv6

Switches running the Network Advantage license support the Hot Standby Router Protocol (HSRP) for IPv6. HSRP provides routing redundancy for routing IPv6 traffic not dependent on the availability of any single router. IPv6 hosts learn of available routers through IPv6 neighbor discovery router advertisement messages. These messages are multicast periodically or are solicited by hosts.

An HSRP IPv6 group has a virtual MAC address that is derived from the HSRP group number and a virtual IPv6 link-local address that is, by default, derived from the HSRP virtual MAC address.

Periodic messages are sent for the HSRP virtual IPv6 link-local address when the HSRP group is active. These messages stop after a final one is sent when the group leaves the active state.



Note When configuring HSRP for IPv6, you must enable HSRP version 2 (HSRPv2) on the interface.

How to Configure HSRP

Default HSRP Configuration

Table 1: Default HSRP Configuration

Feature	Default Setting
HSRP version	Version 1
HSRP groups	None configured
Standby group number	0
Standby MAC address	System assigned as: 0000.0c07.acXX, where XX is the HSRP group number
Standby priority	100
Standby delay	0 (no delay)
Standby track interface priority	10
Standby hello time	3 seconds
Standby holdtime	10 seconds

HSRP Configuration Guidelines

- HSRPv2 and HSRPv1 are mutually exclusive. HSRPv2 is not interoperable with HSRPv1 on an interface and the reverse.
- In the procedures, the specified interface must be one of these Layer 3 interfaces:
 - Routed port: A physical port configured as a Layer 3 port by entering the **no switchport** command in interface configuration mode.
 - SVI: A VLAN interface created by using the **interface vlan** *vlan_id* in global configuration mode, and by default a Layer 3 interface.
 - Etherchannel port channel in Layer 3 mode: A port-channel logical interface created by using the **interface port-channel** *port-channel-number* in global configuration mode, and binding the Ethernet interface into the channel group.
- All Layer 3 interfaces must have IP addresses assigned to them.



Note HSRP millisecond timers are not supported.

Enabling HSRP

The **standby ip** interface configuration command activates HSRP on the configured interface. If an IP address is specified, that address is used as the designated address for the Hot Standby group. If no IP address is specified, the address is learned through the standby function. You must configure at least one Layer 3 port on the LAN with the designated address. Configuring an IP address always overrides another designated address currently in use.

When the **standby ip** command is enabled on an interface and proxy ARP is enabled, if the interface's Hot Standby state is active, proxy ARP requests are answered using the Hot Standby group MAC address. If the interface is in a different state, proxy ARP responses are suppressed.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby version** { 1 | 2 }
4. **standby** [*group-number*] **ip** [*ip-address*] [**secondary**]
5. **end**
6. **show standby** [*interface-id*] [*group*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch(config)# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/1	Enters interface configuration mode, and enter the Layer 3 interface on which you want to enable HSRP.
Step 3	standby version { 1 2 } Example: Switch(config-if)# standby version 1	(Optional) Configures the HSRP version on the interface. <ul style="list-style-type: none"> • 1- Selects HSRPv1. • 2- Selects HSRPv2. <p>If you do not enter this command or do not specify a keyword, the interface runs the default HSRP version, HSRP v1.</p>

	Command or Action	Purpose
Step 4	standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]] Example: Switch(config-if)# standby 1 ip	Creates (or enable) the HSRP group using its number and virtual IP address. <ul style="list-style-type: none"> • (Optional) group-number- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) ip-address- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode
Step 6	show standby [<i>interface-id</i> [<i>group</i>]] Example: Switch # show standby	Verifies the configuration of the standby groups.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring HSRP Priority

The **standby priority**, **standby preempt**, and **standby track** interface configuration commands are all used to set characteristics for finding active and standby routers and behavior regarding when a new active router takes over.

When configuring HSRP priority, follow these guidelines:

- Assigning a priority allows you to select the active and standby routers. If preemption is enabled, the router with the highest priority becomes the active router. If priorities are equal, the current active router does not change.
- The highest number (1 to 255) represents the highest priority (most likely to become the active router).
- When setting the priority, preempt, or both, you must specify at least one keyword (**priority**, **preempt**, or both)

- The priority of the device can change dynamically if an interface is configured with the **standby track** command and another interface on the router goes down.
- The **standby track** interface configuration command ties the router hot standby priority to the availability of its interfaces and is useful for tracking interfaces that are not configured for HSRP. When a tracked interface fails, the hot standby priority on the device on which tracking has been configured decreases by 10. If an interface is not tracked, its state changes do not affect the hot standby priority of the configured device. For each interface configured for hot standby, you can configure a separate list of interfaces to be tracked
- The **standby track interface-priority** interface configuration command specifies how much to decrement the hot standby priority when a tracked interface goes down. When the interface comes back up, the priority is incremented by the same amount.
- When multiple tracked interfaces are down and *interface-priority* values have been configured, the configured priority decrements are cumulative. If tracked interfaces that were not configured with priority values fail, the default decrement is 10, and it is noncumulative.
- When routing is first enabled for the interface, it does not have a complete routing table. If it is configured to preempt, it becomes the active router, even though it is unable to provide adequate routing services. To solve this problem, configure a delay time to allow the router to update its routing table.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP priority characteristics on an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby** [*group-number*] **priority***priority*
4. **standby** [*group-number*] **preempt** [**delay** [*minimumseconds*] [**reloadseconds**] [**syncseconds**]]
5. **standby** [*group-number*] **track** *type number* [*interface-priority*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/1	Enters interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	standby [<i>group-number</i>] priority <i>priority</i> Example: Switch(config-if)# standby 120 priority 50	Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—The group number to which the command applies.

	Command or Action	Purpose
		Use the no form of the command to restore the default values.
Step 4	<p>standby [<i>group-number</i>] preempt [delay [<i>minimumseconds</i>] [reloadseconds] [<i>syncseconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 1 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) group-number—The group number to which the command applies. • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
Step 5	<p>standby [<i>group-number</i>] track <i>type number</i> [<i>interface-priority</i>]</p> <p>Example:</p> <pre>Switch(config-if)# standby track interface gigabitethernet1/1/1</pre>	<p>Configures an interface to track other interfaces so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered.</p> <ul style="list-style-type: none"> • (Optional) group-number— The group number to which the command applies. • type— Enter the interface type (combined with interface number) that is tracked. • number— Enter the interface number (combined with interface type) that is tracked. • (Optional) interface-priority— Enter the amount by which the hot standby priority for the router is decremented or incremented when the interface goes down or comes back up. The default value is 10.
Step 6	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config	Verifies the configuration of the standby groups.

	Command or Action	Purpose
Step 8	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring MHSRP

To enable MHSRP and load-balancing, you configure two routers as active routers for their groups, with virtual routers as standby routers as shown in the *MHSRP Load Sharing* figure in the Multiple HSRP section. You need to enter the **standby preempt** interface configuration command on each HSRP interface so that if a router fails and comes back up, the preemption occurs and restores load-balancing.

Router A is configured as the active router for group 1, and Router B is configured as the active router for group 2. The HSRP interface for Router A has an IP address of 10.0.0.1 with a group 1 standby priority of 110 (the default is 100). The HSRP interface for Router B has an IP address of 10.0.0.2 with a group 2 standby priority of 110.

Group 1 uses a virtual IP address of 10.0.0.3 and group 2 uses a virtual IP address of 10.0.0.4.

Configuring Router A

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>type number</i> Example: Switch (config)# <code>interface gigabitethernet1/0/1</code>	Configures an interface type and enters interface configuration mode.
Step 3	no switchport Example: Switch (config)# <code>no switchport</code>	Switches an interface that is in Layer 2 mode into Layer 3 mode for Layer 3 configuration.

	Command or Action	Purpose
Step 4	<p>ip address <i>ip-address mask</i></p> <p>Example:</p> <pre>Switch (config-if)# ip address 10.0.0.1 255.255.255.0</pre>	Specifies an IP address for an interface.
Step 5	<p>standby [<i>group-number</i>] ip [<i>ip-address</i> [<i>secondary</i>]]</p> <p>Example:</p> <pre>Switch (config-if)# standby 1 ip 10.0.0.3</pre>	<p>Creates the HSRP group using its number and virtual IP address.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 6	<p>standby [<i>group-number</i>] priority <i>priority</i></p> <p>Example:</p> <pre>Switch(config-if)# standby 1 priority 110</pre>	<p>Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—The group number to which the command applies. <p>Use the no form of the command to restore the default values.</p>
Step 7	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>]] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 1 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>-The group number to which the command applies. • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload).

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
Step 8	<p>standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]]</p> <p>Example:</p> <pre>Switch (config-if)# standby 2 ip 10.0.0.4</pre>	<p>Creates the HSRP group using its number and virtual IP address.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 9	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>]] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 2 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>-The group number to which the command applies. • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over).

	Command or Action	Purpose
		Use the no form of the command to restore the default values.
Step 10	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 11	show running-config	Verifies the configuration of the standby groups.
Step 12	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Router B

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface <i>type number</i> Example: Switch (config)# interface gigabitethernet1/0/1	Configures an interface type and enters interface configuration mode.
Step 3	no switchport Example: Switch (config)# no switchport	Switches an interface that is in Layer 2 mode into Layer 3 mode for Layer 3 configuration.
Step 4	ip address <i>ip-address mask</i> Example:	Specifies an IP address for an interface.

	Command or Action	Purpose
	Switch (config-if)# ip address 10.0.0.2 255.255.255.0	
Step 5	<p>standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]]</p> <p>Example:</p> <pre>Switch (config-if)# standby 1 ip 10.0.0.3</pre>	<p>Creates the HSRP group using its number and virtual IP address.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 6	<p>standby [<i>group-number</i>] priority <i>priority</i></p> <p>Example:</p> <pre>Switch(config-if)# standby 2 priority 110</pre>	<p>Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—The group number to which the command applies. <p>Use the no form of the command to restore the default values.</p>
Step 7	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>]] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 1 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>-The group number to which the command applies. • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The

	Command or Action	Purpose
		<p>range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over).</p> <p>Use the no form of the command to restore the default values.</p>
Step 8	<p>standby [<i>group-number</i>] ip [<i>ip-address</i>] [secondary]</p> <p>Example:</p> <pre>Switch (config-if)# standby 2 ip 10.0.0.4</pre>	<p>Creates the HSRP group using its number and virtual IP address.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 9	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>]] [reload <i>seconds</i>] [sync <i>seconds</i>]</p> <p>Example:</p> <pre>Switch(config-if)# standby 2 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>-The group number to which the command applies. • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>

	Command or Action	Purpose
Step 10	end Example: Switch(config-if) # end	Returns to privileged EXEC mode.
Step 11	show running-config	Verifies the configuration of the standby groups.
Step 12	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring HSRP Authentication and Timers

You can optionally configure an HSRP authentication string or change the hello-time interval and hold-time.

When configuring these attributes, follow these guidelines:

- The authentication string is sent unencrypted in all HSRP messages. You must configure the same authentication string on all routers and access servers on a cable to ensure interoperability. Authentication mismatch prevents a device from learning the designated Hot Standby IP address and timer values from other routers configured with HSRP.
- Routers or access servers on which standby timer values are not configured can learn timer values from the active or standby router. The timers configured on an active router always override any other timer settings.
- All routers in a Hot Standby group should use the same timer values. Normally, the *holdtime* is greater than or equal to 3 times the *hellotime*.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP authentication and timers on an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby** [*group-number*] **authentication** *string*
4. **standby** [*group-number*] **timers** *hellotime holdtime*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config) # interface <i>gigabitethernet1/0/1</i>	Enters interface configuration mode, and enter the HSRP interface on which you want to set priority.

	Command or Action	Purpose
Step 3	standby [<i>group-number</i>] authentication <i>string</i> Example: Switch(config-if) # standby 1 authentication word	(Optional) authentication <i>string</i> —Enter a string to be carried in all HSRP messages. The authentication string can be up to eight characters in length; the default string is cisco . (Optional) <i>group-number</i> —The group number to which the command applies.
Step 4	standby [<i>group-number</i>] timers <i>hellotime</i> <i>holdtime</i> Example: Switch(config-if) # standby 1 timers 5 15	(Optional) Configure the time interval to send and receive hello packets. <ul style="list-style-type: none"> • <i>group-number</i>—The group number to which the command applies. • <i>hellotime</i> —Set the interval between successive hello packets in seconds. The range is 1 to 255 seconds. The default is 3. • <i>holdtime</i>—Set the interval to wait for a hello packet from a neighbor device before declaring the neighbor device as inactive. The range is 1 to 255 seconds. The default is 10.
Step 5	end Example: Switch(config-if) # end	Returns to privileged EXEC mode.
Step 6	show running-config	Verifies the configuration of the standby groups.
Step 7	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling HSRP Support for ICMP Redirect Messages

ICMP redirect messages are automatically enabled on interfaces configured with HSRP. ICMP is a network layer Internet protocol that provides message packets to report errors and other information relevant to IP processing. ICMP provides diagnostic functions, such as sending and directing error packets to the host. This feature filters outgoing ICMP redirect messages through HSRP, in which the next hop IP address might be changed to an HSRP virtual IP address. For more information, see the Cisco IOS IP Configuration Guide, Release 12.4.

Configuring HSRP Groups and Clustering

This feature is not supported on the C9500-12Q, C9500-16X, C9500-24Q, C9500-40X models of the Cisco Catalyst 9500 Series Switches. When a device is participating in an HSRP standby routing and clustering is enabled, you can use the same standby group for command switch redundancy and HSRP redundancy. Use the **cluster standby-group** *HSRP-group-name* [**routing-redundancy**] global configuration command to enable the same HSRP standby group to be used for command switch and routing redundancy. If you create a cluster with the same HSRP standby group name without entering the **routing-redundancy** keyword, HSRP standby routing is disabled for the group.

Verifying HSRP

Verifying HSRP Configurations

From privileged EXEC mode, use this command to display HSRP settings:

```
show standby [interface-id [group]] [brief] [detail]
```

You can display HSRP information for the whole switch, for a specific interface, for an HSRP group, or for an HSRP group on an interface. You can also specify whether to display a concise overview of HSRP information or detailed HSRP information. The default display is **detail**. If there are a large number of HSRP groups, using the **show standby** command without qualifiers can result in an unwieldy display.

Example

```
Switch #show standby
VLAN1 - Group 1
Local state is Standby, priority 105, may preempt
Hello time 3 holdtime 10
Next hello sent in 00:00:02.182
Hot standby IP address is 172.20.128.3 configured
Active router is 172.20.128.1 expires in 00:00:09
Standby router is local
Standby virtual mac address is 0000.0c07.ac01
Name is bbb

VLAN1 - Group 100
Local state is Standby, priority 105, may preempt
Hello time 3 holdtime 10
Next hello sent in 00:00:02.262
Hot standby IP address is 172.20.138.51 configured
Active router is 172.20.128.1 expires in 00:00:09
Active router is local
Standby router is unknown expired
Standby virtual mac address is 0000.0c07.ac64
Name is test
```

Configuration Examples for Configuring HSRP

Enabling HSRP: Example

This example shows how to activate HSRP for group 1 on an interface. The IP address used by the hot standby group is learned by using HSRP.



Note This procedure is the minimum number of steps required to enable HSRP. Other configurations are optional.

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby 1 ip
Switch(config-if) # end
Switch # show standby
```

Configuring HSRP Priority: Example

This example activates a port, sets an IP address and a priority of 120 (higher than the default value), and waits for 300 seconds (5 minutes) before attempting to become the active router:

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby ip 172.20.128.3
Switch(config-if) # standby priority 120 preempt delay 300
Switch(config-if) # end
Switch # show standby
```

Configuring MHSRP: Example

This example shows how to enable the MHSRP configuration shown in the figure *MHSRP Load Sharing*

Router A Configuration

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # ip address 10.0.0.1 255.255.255.0
Switch(config-if) # standby ip 10.0.0.3
Switch(config-if) # standby 1 priority 110
Switch(config-if) # standby 1 preempt
Switch(config-if) # standby 2 ip 10.0.0.4
Switch(config-if) # standby 2 preempt
Switch(config-if) # end
```

Router B Configuration

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # ip address 10.0.0.2 255.255.255.0
Switch(config-if) # standby ip 10.0.0.3
Switch(config-if) # standby 1 preempt
Switch(config-if) # standby 2 ip 10.0.0.4
Switch(config-if) # standby 2 priority 110
Switch(config-if) # standby 2 preempt
Switch(config-if) # end
```

Configuring HSRP Authentication and Timer: Example

This example shows how to configure word as the authentication string required to allow Hot Standby routers in group 1 to interoperate:

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby 1 authentication word
Switch(config-if) # end
```

This example shows how to set the timers on standby group 1 with the time between hello packets at 5 seconds and the time after which a router is considered down to be 15 seconds:

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby 1 ip
Switch(config-if) # standby 1 timers 5 15
Switch(config-if) # end
```

Configuring HSRP Groups and Clustering: Example

This example shows how to bind standby group my_hsrp to the cluster and enable the same HSRP group to be used for command switch redundancy and router redundancy. The command can only be executed on the cluster command switch. If the standby group name or number does not exist, or if the switch is a cluster member switch, an error message appears.

```
Switch # configure terminal
Switch(config) # cluster standby-group my_hsrp routing-redundancy
Switch(config-if) # end
```

Additional References for Configuring HSRP

Standards and RFCs

Standard/RFC	Title
<i>RFC 2281</i>	Cisco Hot Standby Router Protocol

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Configuring HSRP

Table 2: Feature Information for Configuring HSRP

Release	Feature Information
Cisco IOS XE Everest 16.5.1a	This feature was introduced.



CHAPTER 2

Configuring NHRP

- [Configuring NHRP, on page 23](#)

Configuring NHRP

The Next Hop Resolution Protocol (NHRP) is an Address Resolution Protocol (ARP)-like protocol that dynamically maps a nonbroadcast multiaccess (NBMA) network, instead of manually configuring all the tunnel end points. With NHRP, systems attached to an NBMA network can dynamically learn the NBMA (physical) address of the other systems that are part of that network, allowing these systems to directly communicate. This protocol provides an ARP-like solution which allows stations' data-link addresses to be dynamically determined.

NHRP is a client and server protocol where the hub is the Next Hop Server (NHS) and the spokes are the Next Hop Clients (NHCs). The hub maintains an NHRP database of the public interface addresses of each spoke. Each spoke registers its non-NBMA (real) address when it boots and queries the NHRP database for addresses of the destination spokes to build direct tunnels.

This module explains how to configure NHRP with generic routing encapsulation (GRE). In Cisco IOS XE Denali 16.3.1, the NHRP supports only spoke configurations.

Information About Configuring NHRP

NHRP and NBMA Network Interaction

Most WAN networks are a collection of point-to-point links. Virtual tunnel networks (for example Generic Routing Encapsulation [GRE] tunnels) are also a collection of point-to-point links. To effectively scale the connectivity of these point-to-point links, they are usually grouped into a single or multilayer hub-and-spoke network. Multipoint interfaces (for example, GRE tunnel interfaces) can be used to reduce the configuration on a hub router in such a network. This resulting network is a NBMA network.

Because there are multiple tunnel endpoints that are reachable through a single multipoint interface, there needs to be a mapping from the logical tunnel endpoint IP address to the physical tunnel endpoint IP address, to forward packets out of the tunnel interfaces over this NBMA network. This mapping could be statically configured, but it is preferable if the mapping can be discovered or learned dynamically.

NHRP is an ARP-like protocol that alleviates these NBMA network problems. With NHRP, systems attached to an NBMA network dynamically learn the NBMA address of other systems that are part of the network, allowing these systems to directly communicate without requiring traffic to use an intermediate hop.

Routers, access servers, and hosts can use NHRP to discover the addresses of other routers and hosts connected to an NBMA network. Partially-meshed NBMA networks typically have multiple logical networks behind the NBMA network. In such configurations, packets traversing the NBMA network might have to make several hops over the NBMA network before arriving at the exit router (the router nearest the destination network).

NHRP Registration helps support these NBMA networks:

- **NHRP Registration**—NHRP allows Next Hop Clients (NHCs) to dynamically register with Next Hop Servers (NHSs). This registration function allows the NHCs to join the NBMA network without configuration changes on the NHSs, especially in cases where the NHC has a dynamic physical IP address or is behind a Network Address Translation (NAT) router that dynamically changes the physical IP address. In these cases, it would be impossible to preconfigure the logical (VPN IP address) to physical (NBMA IP) mapping for the NHC on the NHS.

Dynamically Built Hub-and-Spoke Networks

With NHRP, the NBMA network is initially laid out as a hub-and-spoke network that can have multiple hierarchical layers of NHCs as spokes and NHSs as hubs. The NHCs are configured with static mapping information to reach their NHSs and will connect to their NHS and send an NHRP registration to the NHS. This configuration allows the NHS to dynamically learn the mapping information for the spoke, reducing the configuration needed on the hub and allowing the spoke to obtain a dynamic NBMA (physical) IP address.

How to Configure NHRP

Enabling NHRP on an Interface

Perform this task to enable NHRP for an interface on a switch. In general, all NHRP stations within a logical NBMA network should be configured with the same network identifier.

The NHRP network ID is used to define the NHRP domain for an NHRP interface and differentiate between multiple NHRP domains or networks, when two or more NHRP domains (GRE tunnel interfaces) are available on the same NHRP node (switch). The NHRP network ID helps keep two NHRP networks (clouds) separate when both are configured on the same switch.

The NHRP network ID is a local-only parameter. It is significant only to the local switch and is not transmitted in NHRP packets to other NHRP nodes. For this reason the actual value of the NHRP network ID configured on a switch need not match the same NHRP network ID on another switch where both of these switches are in the same NHRP domain. As NHRP packets arrive on a GRE interface, they are assigned to the local NHRP domain in the NHRP network ID that is configured on that interface.

We recommend that the same NHRP network ID be used on the GRE interfaces on all switches that are in the same NHRP network. It is then easier to track which GRE interfaces are members of which NHRP network.

NHRP domains (network IDs) can be unique on each GRE tunnel interface on a switch. NHRP domains can span across GRE tunnel interfaces on a route. In this case the effect of using the same NHRP network ID on the GRE tunnel interfaces is to merge the two GRE interfaces into a single NHRP network.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*

4. **ip address** *ip-address network-mask*
5. **ip nhrp network-id** *number*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: <pre>Switch(config)# interface tunnel 100</pre>	Configures an interface and enters interface configuration mode.
Step 4	ip address <i>ip-address network-mask</i> Example: <pre>Switch(config-if)# ip address 10.0.0.1 255.255.255.0</pre>	Enables IP and gives the interface an IP address.
Step 5	ip nhrp network-id <i>number</i> Example: <pre>Switch(config-if)# ip nhrp network-id 1</pre>	Enables NHRP on the interface.
Step 6	end Example: <pre>Switch(config)# end</pre>	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring a GRE Tunnel for Multipoint Operation

Perform this task to configure a GRE tunnel for multipoint (NMBA) operation.

A tunnel network of multipoint tunnel interfaces can be considered of as an NBMA network. When multiple GRE tunnels are configured on the same switch, they must either have unique tunnel ID keys or unique tunnel source addresses.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address*
5. **ip mtu** *bytes*
6. **ip pim sparse-dense-mode**
7. **ip nhrp map** *ip-address nbma-address*
8. **ip nhrp map multicast** *nbma-address*
9. **ip nhrp network-id** *number*
10. **ip nhrp nhs** *nhs-address*
11. **tunnel source vlan** *interface-number*
12. **tunnel destination** *ip-address*
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Switch(config)# interface tunnel 100	Configures an interface and enters interface configuration mode.
Step 4	ip address <i>ip-address</i> Example: Switch(config-if)# ip address 172.16.1.1 255.255.255.0	Configures an IP address for the interface.
Step 5	ip mtu <i>bytes</i> Example: Switch(config-if)# ip mtu 1400	Sets the maximum transmission unit (MTU) size of IP packets sent on an interface.
Step 6	ip pim sparse-dense-mode Example: Switch(config-if)# ip pim sparse-dense-mode	Enables Protocol Independent Multicast (PIM) on an interface and treats the interface in either sparse mode or dense mode of operation, depending on which mode the multicast group operates in.

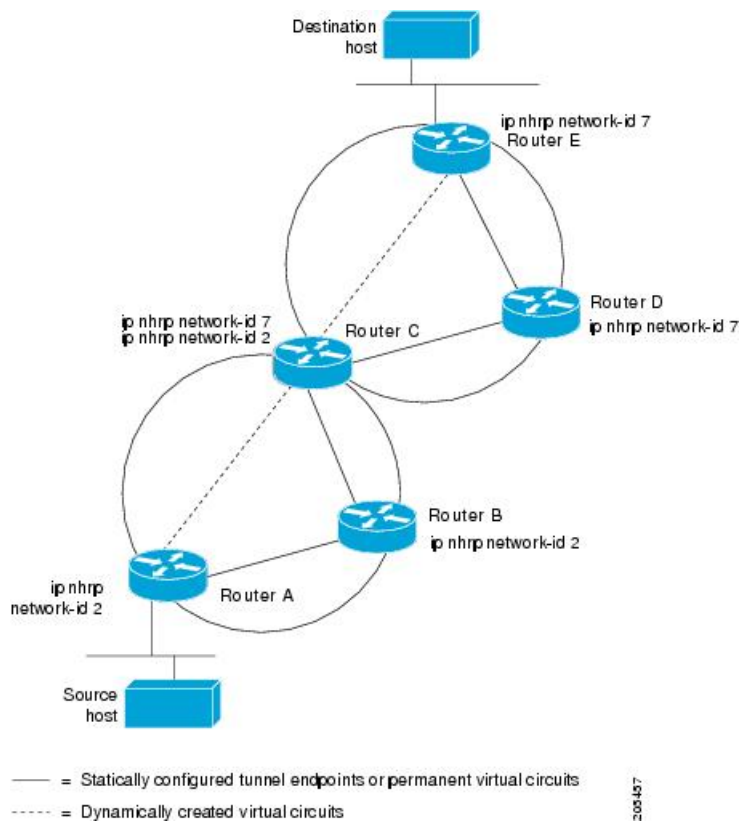
	Command or Action	Purpose
Step 7	<p>ip nhrp map <i>ip-address nbma-address</i></p> <p>Example:</p> <pre>Switch(config-if)# ip nhrp map 172.16.1.2 10.10.10.2</pre>	<p>Statically configures the IP-to-nonbroadcast multiaccess (NBMA) address mapping of IP destinations connected to an NBMA network.</p> <ul style="list-style-type: none"> • <i>ip-address</i>—IP address of the destinations reachable through the NBMA network. This address is mapped to the NBMA address. • <i>nbma-address</i>—NBMA address that is directly reachable through the NBMA network. The address format varies depending on the medium used. For example, ATM has a Network Service Access Point (NSAP) address, Ethernet has a MAC address, and Switched Multimegabit Data Service (SMDS) has an E.164 address. This address is mapped to the IP address.
Step 8	<p>ip nhrp map multicast <i>nbma-address</i></p> <p>Example:</p> <pre>Switch(config-if)# ip nhrp map multicast 10.10.10.2</pre>	<p>Configures nonbroadcast multiaccess (NBMA) addresses used as destinations for broadcast or multicast packets to be sent over a tunnel network.</p>
Step 9	<p>ip nhrp network-id <i>number</i></p> <p>Example:</p> <pre>Switch(config-if)# ip nhrp network-id 1</pre>	<p>Enable the Next Hop Resolution Protocol (NHRP) on an interface.</p> <ul style="list-style-type: none"> • <i>number</i>—Globally unique, 32-bit network ID from a nonbroadcast multiaccess (NBMA) network. The range is from 1 to 4294967295.
Step 10	<p>ip nhrp nhs <i>nhs-address</i></p> <p>Example:</p> <pre>Switch(config-if)# ip nhrp nhs 172.16.1.2</pre>	<p>Specifies the address of one or more NHRP servers.</p> <ul style="list-style-type: none"> • <i>nhs-address</i>—Address of the next-hop server being specified.
Step 11	<p>tunnel source vlan <i>interface-number</i></p> <p>Example:</p> <pre>Switch(config-if)# tunnel source vlan 1</pre>	<p>Sets the source address for a tunnel interface</p>
Step 12	<p>tunnel destination <i>ip-address</i></p> <p>Example:</p> <pre>Switch(config-if)# tunnel destination 10.10.10.2</pre>	<p>Sets the destination address for a tunnel interface.</p>
Step 13	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	<p>Exits interface configuration mode and returns to privileged EXEC mode.</p>

Configuration Examples for NHRP

Physical Network Designs for Logical NBMA Examples

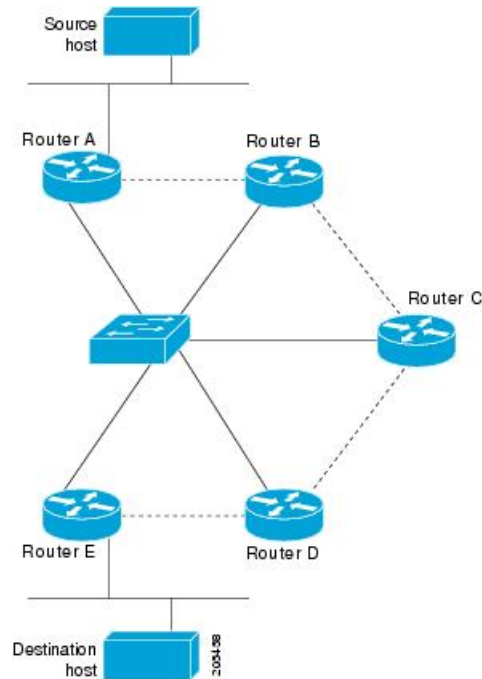
A logical NBMA network is considered the group of interfaces and hosts participating in NHRP and having the same network identifier. The figure below illustrates two logical NBMA networks (shown as circles) configured over a single physical NBMA network. Router A can communicate with routers B and C because they share the same network identifier (2). Router C can also communicate with routers D and E because they share network identifier 7. After address resolution is complete, router A can send IP packets to router C in one hop, and router C can send them to router E in one hop, as shown by the dotted lines.

Figure 3: Two Logical NBMA Networks over One Physical NBMA Network



The physical configuration of the five routers in the figure above might actually be that shown in the figure below. The source host is connected to router A and the destination host is connected to router E. The same switch serves all five routers, making one physical NBMA network.

Figure 4: Physical Configuration of a Sample NBMA Network



Refer again to the first figure above. Initially, before NHRP has resolved any NBMA addresses, IP packets from the source host to the destination host travel through all five routers connected to the switch before reaching the destination. When router A first forwards the IP packet toward the destination host, router A also generates an NHRP request for the IP address of the destination host. The request is forwarded to router C, whereupon a reply is generated. Router C replies because it is the egress router between the two logical NBMA networks.

Similarly, router C generates an NHRP request of its own, to which router E replies. In this example, subsequent IP traffic between the source and the destination still requires two hops to traverse the NBMA network, because the IP traffic must be forwarded between the two logical NBMA networks. Only one hop would be required if the NBMA network were not logically divided.

Example: GRE Tunnel for Multipoint Operation

With multipoint tunnels, a single tunnel interface may be connected to multiple neighboring switches. Unlike point-to-point tunnels, a tunnel destination need not be configured. In fact, if configured, the tunnel destination must correspond to an IP multicast address.

In the following example, switches A and B share an Ethernet segment. Minimal connectivity over the multipoint tunnel network is configured, thus creating a network that can be treated as a partially meshed NBMA network. Due to the static NHRP map entries, switch A knows how to reach switch B and vice versa.

The following example shows how to configure a GRE multipoint tunnel:

Switch A Configuration

```
Switch(config)# interface tunnel 100 !Tunnel interface configured for PIM traffic
Switch(config-if)# no ip redirects
Switch(config-if)# ip address 192.168.24.1 255.255.255.252
Switch(config-if)# ip mtu 1400
```

```
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip nhrp map 192.168.24.3 172.16.0.1 !NHRP may optionally be configured
to dynamically discover tunnel end points.
Switch(config-if)# ip nhrp map multicast 172.16.0.1
Switch(config-if)# ip nhrp network-id 1
Switch(config-if)# ip nhrp nhs 192.168.24.3
Switch(config-if)# tunnel source vlan 1
Switch(config-if)# tunnel destination 172.16.0.1
Switch(config-if)# end
```

Switch B Configuration

```
Switch(config)# interface tunnel 100
Switch(config-if)# no ip redirects
Switch(config-if)# ip address 192.168.24.2 255.255.255.252
Switch(config-if)# ip mtu 1400
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip nhrp map 192.168.24.4 10.10.0.3
Switch(config-if)# ip nhrp map multicast 10.10.10.3
Switch(config-if)# ip nhrp network-id 1
Switch(config-if)# ip nhrp nhs 192.168.24.4
Switch(config-if)# tunnel source vlan 1
Switch(config-if)# tunnel destination 10.10.10.3
Switch(config-if)# end
```

Additional References for Configuring NHRP

RFCs

RFC	Title
RFC 2332	NBMA Next Hop Resolution Protocol (NHRP)

Feature Information for Configuring NHRP

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 Configuring NHRP

Feature Name	Releases	Feature Information
Next Hop Resolution Protocol	Cisco IOS XE Polaris 16.3.1	The Next Hop Resolution Protocol (NHRP) is an Address Resolution Protocol (ARP)-like protocol that dynamically maps a nonbroadcast multiaccess (NBMA) network instead of manually configuring all the tunnel end points. With NHRP, systems attached to an NBMA network can dynamically learn the NBMA (physical) address of the other systems that are part of that network, allowing these systems to directly communicate.
	Cisco IOS XE Everest 16.5.1a	This feature was implemented on the following platforms: <ul style="list-style-type: none"> • Cisco Catalyst 3650 Series Switches • Cisco Catalyst 3850 Series Switches • Cisco Catalyst 9300 Series Switches • Cisco Catalyst 9500 Series Switches



CHAPTER 3

Configuring Network Address Translation

- Finding Feature Information, on page 33
- Network Address Translation (NAT), on page 34
- Benefits of Configuring NAT, on page 34
- How NAT Works, on page 34
- Uses of NAT, on page 35
- NAT Inside and Outside Addresses, on page 35
- Types of NAT, on page 36
- Using NAT to Route Packets to the Outside Network (Inside Source Address Translation), on page 37
- Outside Source Address Translation, on page 38
- Port Address Translation (PAT), on page 38
- Overlapping Networks, on page 40
- Limitations of NAT, on page 41
- Performance and Scale Numbers for NAT, on page 42
- Address Only Translation, on page 42
- Restrictions for Address Only Translation, on page 42
- Configuring NAT, on page 43
- Using Application-Level Gateways with NAT, on page 55
- Best Practices for NAT Configuration, on page 55
- Troubleshooting NAT, on page 56
- Feature Information for Network Address Translation, on page 56

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Network Address Translation (NAT)

Network Address Translation (NAT) is designed for IP address conservation. It enables private IP networks that use unregistered IP addresses to connect to the Internet. NAT operates on a device, usually connecting two networks together, and translates the private (not globally unique) addresses in the internal network into global routable 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. This ability provides additional security by effectively hiding the entire internal network behind that one address. NAT offers the dual functions of security and address conservation and is typically implemented in remote-access environments.

NAT is also used at the enterprise edge to allow internal users access to the Internet and to allow Internet access to internal devices such as mail servers.

Benefits of Configuring NAT

- Resolves the problem of IP depletion.

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 yet possess Network Information Center (NIC)-registered IP addresses must acquire IP addresses, and if more than 254 clients are present or are planned, the scarcity of Class B addresses becomes a serious issue. NAT addresses these issues by mapping thousands of hidden internal addresses to a range of easy-to-get Class C addresses.

- Provides a layer of security by preventing the client IP address from being exposed to the outside network.

Sites that already have registered IP addresses for clients on an internal network may want to hide those addresses from the Internet so that hackers cannot directly attack clients. With client addresses hidden, a degree of security is established. NAT gives LAN administrators complete freedom to expand Class A addressing, which is drawn from the reserve pool of the Internet Assigned Numbers Authority. The expansion of Class A addresses occurs within the organization without a concern for addressing changes at the LAN or the Internet interface.

- Cisco software can selectively or dynamically perform NAT. This flexibility allows network administrator to use RFC 1918 addresses or registered addresses.
- NAT is designed for use on a variety of devices for IP address simplification and conservation. In addition, NAT allows the selection of internal hosts that are available for translation.
- A significant advantage of NAT is that it can be configured without requiring any changes to devices other than to those few devices on which NAT will be configured.

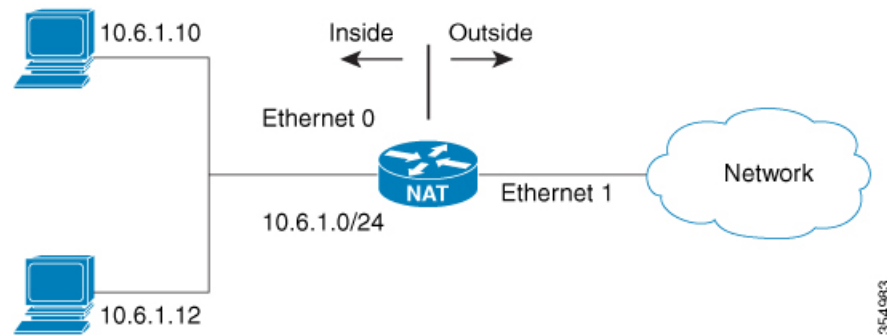
How NAT Works

A device that is 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 device 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. Multiple inside networks could be connected to the device and similarly there

might exist multiple exit points from the device towards outside networks. If NAT cannot allocate an address because it has run out of addresses, it drops the packet and sends an Internet Control Message Protocol (ICMP) host unreachable packet to the destination.

Translation and forwarding are performed in the hardware switching plane, thereby improving the overall throughput performance. For more details on performance, refer the section on [Performance and Scale Numbers for NAT](#)

Figure 5: NAT



Uses of NAT

NAT can be used for the following scenarios:

- To connect to the Internet when only a few of your hosts have globally unique IP address.

NAT is configured on a device at the border of a stub domain (referred to as the inside network) and a public network such as the Internet (referred to as the outside network). NAT translates internal local addresses to globally unique IP addresses before sending packets to the outside network. As a solution to the connectivity problem, NAT is practical only when relatively few hosts in a stub domain communicate outside of the domain at the same time. When this is the case, only a small subset of the IP addresses in the domain must be translated into globally unique IP addresses when outside communication is necessary, and these addresses can be reused

- Renumbering:

Instead of changing the internal addresses, which can be a considerable amount of work, you can translate them by using NAT.

NAT Inside and Outside Addresses

The term *inside* in a NAT context refers to networks owned by an organization that must be translated. When NAT is configured, hosts within this network will have addresses in one space (known as the local address space) that will appear to those outside the network as being in another space (known as the global address space).

Similarly, the term *outside* refers to those networks to which the stub network connects, and which are generally not under the control of an organization. Hosts in outside networks can also be subject to translation, and can thus have local and global addresses.

NAT uses the following definitions:

- Inside local address—an IP address that is assigned to a host on the inside network. The address is probably not a routable IP address assigned by NIC or service provider.
- Inside global address—a global routable IP address (assigned by the NIC or service provider) that represents one or more inside local IP addresses to the outside world.
- Outside local address—the IP address of an outside host as it appears to the inside network. Not necessarily a routable IP address, it is allocated from the address space that is routable on the inside.
- Outside global address—the IP address assigned to a host on the outside network by the owner of the host. The address is allocated from a globally routable address or network space.
- Inside Source Address Translation—translates an inside local address to inside global address.
- Outside Source Address Translation—translates the outside global address to outside local address.
- Static Port Translation—translates the IP address and port number of an inside/outside local address to the IP address and port number of the corresponding inside/outside global address.
- Static Translation of a given subnet—translates a specified range of subnets of an inside/outside local address to the corresponding inside/outside global address.
- Half Entry—represents a mapping between the local and global address/ports and is maintained in the translation database of NAT module. A half entry may be created statically or dynamically based on the configured NAT rule.
- Full Entry/Flow entry—represents a unique flow corresponding to a given session. In addition to the local to global mapping, it also maintains the destination information which fully qualifies the given flow. A Full entry is always created dynamically and maintained in the translation database of NAT module.

Types of NAT

You can configure NAT such that it will advertise only a single address for your entire network to the outside world. Doing this effectively hides the internal network from the world, giving you some additional security.

The types of NAT include:

- Static address translation (static NAT)—Allows one-to-one mapping between local and global addresses.
- Dynamic address translation (dynamic NAT)—Maps unregistered IP addresses to registered IP addresses from a pool of registered IP addresses.
- Overloading / PAT—Maps multiple unregistered IP addresses to a single registered IP address (many to one) using different Layer 4 ports. This method is also known as Port Address Translation (PAT). By using overloading, thousands of users can be connected to the Internet by using only one real global IP address.

Using NAT to Route Packets to the Outside Network (Inside Source Address Translation)

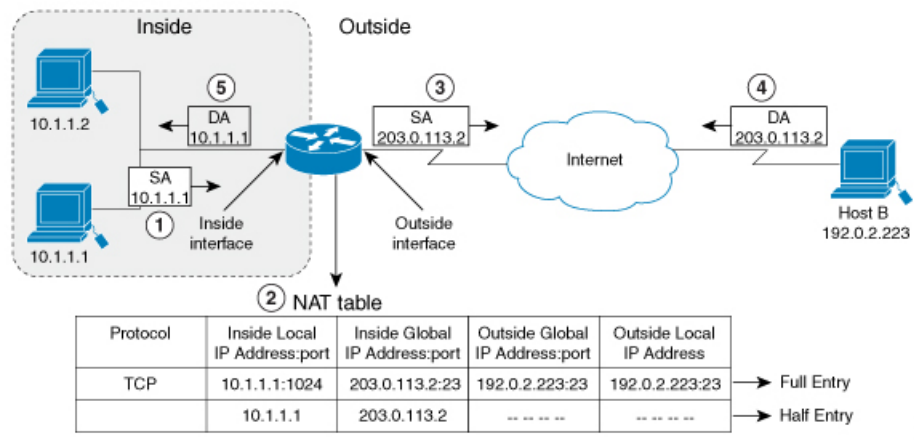
You can translate unregistered IP addresses into globally unique IP addresses when communicating outside your network.

You can configure static or dynamic inside source address translation as follows:

- Static translation establishes a one-to-one mapping between the inside local address and an inside global address. Static translation is useful when a host on the inside must be accessible by a fixed address from the outside. Static translation can be enabled by configuring a static NAT rule as explained in the [Configuring Static Translation of Inside Source Addresses, on page 43](#) section.
- Dynamic translation establishes a mapping between an inside local address and a pool of global addresses dynamically. Dynamic translation can be enabled by configuring a dynamic NAT rule and the mapping is established based on the result of the evaluation of the configured rule at run-time. You can employ an Access Control List (ACL), both Standard and Extended ACLs, to specify the inside local address. The inside global address can be specified through an address pool or an interface. Dynamic translation is enabled by configuring a dynamic rule as explained in the [Configuring Dynamic Translation of Inside Source Addresses](#) section.

The figure below illustrates a device that is translating a source address inside a network to a source address outside the network.

Figure 6: NAT Inside Source Translation



The following process describes the inside source address translation, as shown in the figure above:

1. The user at host 10.1.1.1 opens a connection to Host B in the outside network.
2. NAT module intercepts the corresponding packet and attempts to translate the packet.

The following scenarios are possible based on the presence or absence of a matching NAT rule:

- If a matching static translation rule exists, the packet gets translated to the corresponding inside global address. Otherwise, the packet is matched against the dynamic translation rule and in the event of a successful match, it gets translated to the corresponding inside global address. The NAT module

inserts a fully qualified flow entry corresponding to the translated packet, into its translation database. This facilitates fast translation and forwarding of the packets corresponding to this flow, in either direction.

- The packet gets forwarded without any address translation in the absence of a successful rule match.
- The packet gets dropped in the event of failure to obtain a valid inside global address even-though we have a successful rule match.



Note If an ACL is employed for dynamic translation, NAT evaluates the ACL and ensures that only the packets that are permitted by the given ACL are considered for translation.

3. The device replaces the inside local source address of host 10.1.1.1 with the inside global address of the translation, 203.0.113.2, and forwards the packet.
4. Host B receives the packet and responds to host 10.1.1.1 by using the inside global IP destination address (DA) 203.0.113.2
5. The response packet from host B would be destined to the inside global address and the NAT module intercepts this packet and translates it back to the corresponding inside local address with the help of the flow entry that has been setup in the translation database.

Host 10.1.1.1 receives the packet and continues the conversation. The device performs Steps 2 to 5 for each packet that it receives.

Outside Source Address Translation

You can translate the source address of the IP packets that travel from outside of the network to inside the network. This type of translation is usually employed in conjunction with inside source address translation to interconnect overlapping networks.

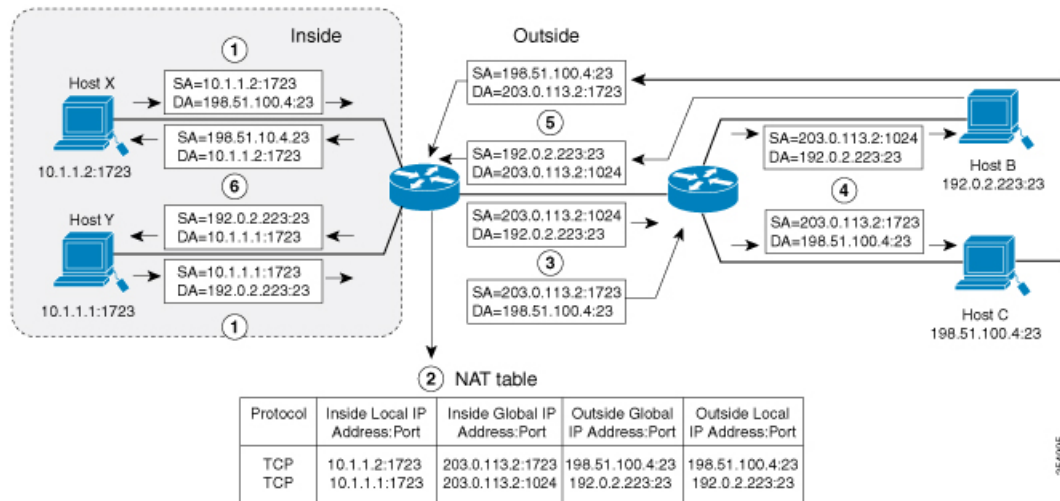
This process is explained in the section on [Configuring Translation of Overlapping Networks, on page 50](#)

Port Address Translation (PAT)

You can conserve addresses in the inside global address pool by allowing a device to use one global address for many local addresses and this type of NAT configuration is called overloading or port address translation. When overloading is configured, the device maintains enough information from higher-level protocols (for example, TCP or UDP port numbers) to translate the global address back to the correct local address. When multiple local addresses map to one global address, the TCP or UDP port numbers of each inside host distinguish between the local addresses.

The figure below illustrates a NAT operation when an inside global address represents multiple inside local addresses. The TCP port numbers act as differentiators.

Figure 7: PAT / NAT Overloading Inside Global Addresses



The device performs the following process in the overloading of inside global addresses, as shown in the figure above. Both Host B and Host C believe that they are communicating with a single host at address 203.0.113.2. Whereas, they are actually communicating with different hosts; the port number is the differentiator. In fact, many inside hosts can share the inside global IP address by using many port numbers.

1. The user at host 10.1.1.1:1723 opens a connection to Host B and the user at host 10.1.1.2:1723 opens a connection to Host C.
2. NAT module intercepts the corresponding packets and attempts to translate the packets.

Based on the presence or absence of a matching NAT rule the following scenarios are possible:

- If a matching static translation rule exists, then it takes precedence and the packets are translated to the corresponding global address. Otherwise, the packets are matched against dynamic translation rule and in the event of a successful match, they are translated to the corresponding global address. NAT module inserts a fully qualified flow entry corresponding to the translated packets, into its translation database, to facilitate fast translation and forwarding of the packets corresponding to this flow, in either direction.
 - The packets get forwarded without any address translation in the absence of a successful rule match.
 - The packets get dropped in the event of failure to obtain a valid inside global address even though we have a successful rule match.
 - As this is a PAT configuration, transport ports help translate multiple flows to a single global address. (In addition to source address, the source port is also subjected to translation and the associated flow entry maintains the corresponding translation mappings.)
3. The device replaces inside local source address/port 10.1.1.1/1723 and 10.1.1.2/1723 with the corresponding selected global address/port 203.0.113.2/1024 and 203.0.113.2/1723 respectively and forwards the packets.
 4. Host B receives the packet and responds to host 10.1.1.1 by using the inside global IP address 203.0.113.2, on port 1024. Host C receives the packet and responds to host 10.1.1.2 using the inside global IP address 203.0.113.2, on port 1723.

- When the device receives the packets with the inside global IP address, it performs a NAT table lookup; the inside global address and port, and the outside address and port as keys; translates the addresses to the inside local addresses 10.1.1.1:1723 / 10.1.1.2:1723 and forwards the packets to host 10.1.1.1. and 10.1.1.2 respectively.

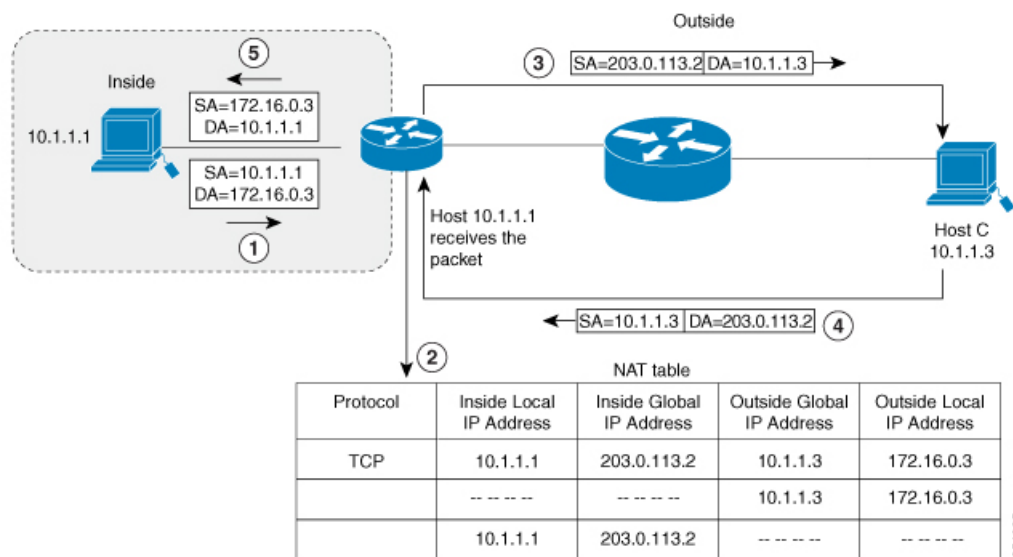
Host 10.1.1.1 and Host 10.1.1.2 receive the packet and continue the conversation. The device performs Steps 2 to 5 for each packet it receives.

Overlapping Networks

Use NAT to translate IP addresses if the IP addresses that you use are neither legal nor officially assigned. Overlapping networks result when you assign an IP address to a device on your network that is already legally owned and assigned to a different device on the Internet or outside the network.

The following figure depicts overlapping networks: the inside network and outside network both have the same local IP addresses (10.1.1.x). You need network connectivity between such overlapping address spaces with one NAT device to translate the address of a remote peer (10.1.1.3) to a different address from the perspective of the inside.

Figure 8: NAT Translating Overlapping Addresses



Notice that the inside local address (10.1.1.1) and the outside global address (10.1.1.3) are in the same subnet. To translate the overlapping address, first, the inside source address translation happens with the inside local address getting translated to 203.0.113.2 and a half entry is created in the NAT table. On the Receiving side, the outside source address is translated to 172.16.0.3 and another half entry is created. The NAT table is then updated with a full entry of the complete translation.

The following steps describe how a device translates overlapping addresses:

- Host 10.1.1.1 opens a connection to 172.16.0.3.
- The NAT module sets up the translation mapping of the inside local and global addresses to each other and the outside global and local addresses to each other

3. The Source Address (SA) is replaced with inside global address and the Destination Address (DA) is replaced with outside global address.
4. Host C receives the packet and continues the conversation.
5. The device does a NAT table lookup, replaces the DA with inside local address, and replaces the SA with outside local address.
6. Host 10.1.1.1 receives the packet and the conversation continues using this translation process.

Limitations of NAT

- There are certain NAT operations that are currently not supported in the Hardware data plane. The following are such operations that are carried out in the relatively slower Software data plane:
 - Translation of Internet Control Message Protocol (ICMP) packets.
 - Translation of packets that require application layer gateway (ALG) processing.
 - Packets that require both inside and outside translation.



Note This does not apply to the Cisco Catalyst 9500 Series Switches.

- The maximum number of sessions that can be translated and forwarded in the hardware in an ideal setting is limited to 7000 for the Cisco Catalyst 9500 Series Switches and 7750 for the Cisco Catalyst 9500 High Performance Series Switches. For the Cisco Catalyst 9500 Series Switches, maximum number of sessions that can be translated is based on SDM template configured. Additional flows that require translation are handled in the software data plane at a reduced throughput.



Note Each translation consumes two entries in TCAM.

- A configured NAT rule might fail to get programmed into the hardware owing to resource constraint. This could result in packets that correspond to the given rule to get forwarded without translation.



Note This does not apply to the Cisco Catalyst 9500 Series Switches.

- ALG support is currently limited to FTP, TFTP and ICMP protocols. Also, although TCP SYN, TCP FIN and TCP RST are not part of ALG traffic, they are processed as part of ALG traffic.
- Dynamically created NAT flows age out after a period of inactivity. For the Cisco Catalyst 9500 Series Switches, the number of NAT flows whose activity can be tracked depends on the SDM template configured. For example, the number of NAT flows whose activity can be tracked is limited to 14000 when NAT template is configured. As a result, it is possible that certain active NAT flows get prematurely removed in cases where the total number of NAT flows exceeds the maximum limit. Also, attributes such as *used time* and *remaining time* associated with the NAT flows might report incorrect values in such cases.

- For NAT traffic, the CPU Queue bandwidth limitation is 2000 packets. Packets that exceed this limit will be dropped.
- Port Channel is not supported in NAT configuration.
- NAT does not support translation of fragmented packets.
- Bidirectional Forwarding Detection (BFD) is not supported with NAT configuration.
- NAT configuration must be done without using route-maps, as route-mapped NAT is not supported.
- Explicit deny access control entry (ACE) in NAT ACL is not supported. Only explicit permit ACE is supported.

Performance and Scale Numbers for NAT

Configure SDM template NAT to achieve better performance and scale number. Refer [Configuring Switch Database Management \(SDM\) Template, on page 54](#)

The maximum number of TCAM flows that are available in the hardware is 14000 for the Cisco Catalyst 9500 Series Switch and 15500 for the Cisco Catalyst 9500 High Performance Series Switches.



Note Using Address Only Translation optimizes the handling of flows and enhances the scale of the NAT feature.

Address Only Translation

Address only Translation (AOT) functionality can be employed in situations that require only the address fields to be translated and not the transport ports. In such settings, enabling AOT functionality significantly increases the number of flows that can be translated and forwarded in the hardware at line-rate. This improvement is brought about by optimizing the usage of various hardware resources associated with translation and forwarding. A typical NAT focused resource allocation scheme sets aside 14000 (Cisco Catalyst 9500 Series Switch) and 15500 (Cisco Catalyst 9500 High Performance Series Switches) TCAM entries for performing hardware translation. This places a strict upper limit on the number of flows that can be translated and forwarded at line-rate. Under AOT scheme, the usage of TCAM resource is highly optimized thereby enabling the accommodation of more number of flows in the TCAM tables and this provides a significant improvement in the hardware translation and forwarding scale. AOT can be very effective in situations where majority of the flows are destined to a single or a small set of destinations. Under such favourable conditions, AOT can potentially enable line-rate translation and forwarding of all the flows originating from the given end-point(s). AOT functionality is disabled by default. It can be enabled using the **no ip nat create flow-entries** command. The existing dynamic flow can be cleared using the **clear ip nat translation** command. The AOT feature can be disabled using the **ip nat create flow-entries** command.

Restrictions for Address Only Translation

- AOT feature is expected to function correctly only in translation scenarios corresponding to simple inside static and inside dynamic rules

- When AOT is enabled, the **show ip nat translation** command will not give visibility into all the NAT flows being translated and forwarded.

Configuring NAT

The tasks described in this section will help you configure NAT. Based on the desired configuration, you may need to configure more than one task.

Configuring Static Translation of Inside Source Addresses

Configure static translation of inside source address to allow one-to-one mapping between an inside local address and an inside global address. Static translation is useful when a host on the inside must be accessible by a fixed address from the outside.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Use any of the following three commands depending on the requirement:
 - **ip nat inside source static** *local-ip global-ip*

```
Switch(config)# ip nat inside source static 10.10.10.1 172.16.131.1
```
 - **ip nat inside source static** *protocol local-ip port global-ip port*

```
Switch(config)# ip nat inside source static tcp 10.10.10.1 1234 172.16.131.1 5467
```
 - **ip nat inside source static network** *local-ip global-ip {prefix_len len | subnet subnet-mask}*

```
Switch(config)# ip nat inside source static network 10.10.10.1 172.16.131.1
prefix_len 24
```
4. **interface** *type number*
5. **ip address** *ip-address mask* [**secondary**]
6. **ip nat inside**
7. **exit**
8. **interface** *type number*
9. **ip address** *ip-address mask* [**secondary**]
10. **ip nat outside**
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	Use any of the following three commands depending on the requirement: <ul style="list-style-type: none"> • ip nat inside source static <i>local-ip global-ip</i> Switch(config)# ip nat inside source static 10.10.10.1 172.16.131.1. • ip nat inside source static <i>protocol local-ip port global-ip port</i> Switch(config)# ip nat inside source static tcp 10.10.10.1 1234 172.16.131.1 5467 • ip nat inside source static network <i>local-ip global-ip {prefix_len len subnet subnet-mask}</i> Switch(config)# ip nat inside source static network 10.10.10.1 172.16.131.1 prefix_len 24 	Establishes static translation between an inside local address and an inside global address. Establishes a static port translation between an inside local address and an inside global address. Establishes a static translation between an inside local address and an inside global address. You can specify a range of subnets to be translated to the inside global address, wherein the host portion of the IP address gets translated and the network portion of the IP remains the same.
Step 4	interface <i>type number</i> Example: Switch(config)# interface ethernet 1	Specifies an interface and enters interface configuration mode.
Step 5	ip address <i>ip-address mask [secondary]</i> Example: Switch(config-if)# ip address 10.114.11.39 255.255.255.0	Sets a primary IP address for an interface.
Step 6	ip nat inside Example: Switch(config-if)# ip nat inside	Connects the interface to the inside network, which is subject to NAT.
Step 7	exit Example: Switch(config-if)# exit	Exits interface configuration mode and returns to global configuration mode.
Step 8	interface <i>type number</i> Example: Switch(config)# interface gigabitethernet 0/0/0	Specifies a different interface and enters interface configuration mode.
Step 9	ip address <i>ip-address mask [secondary]</i> Example: Switch(config-if)# ip address 172.31.232.182 255.255.255.240	Sets a primary IP address for an interface.

	Command or Action	Purpose
Step 10	ip nat outside Example: Switch(config-if)# ip nat outside	Connects the interface to the outside network.
Step 11	end Example: Switch(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring Dynamic Translation of Inside Source Addresses

Dynamic translation establishes a mapping between an inside local address and a pool of global addresses dynamically. Dynamic translation can be enabled by configuring a dynamic NAT rule and the mapping is established based on the result of the evaluation of the configured rule at run-time. You can employ an ACL to specify the inside local address and the inside global address can be specified through an address pool or an interface.

Dynamic translation is useful when multiple users on a private network need to access the Internet. The dynamically configured pool IP address may be used as needed and is released for use by other users when access to the internet is no longer required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip nat pool** *name start-ip end-ip netmask netmask* | **prefix-length** *prefix-length*
4. **access-list** *access-list-number permit source [source-wildcard]*
5. **ip nat inside source list** *access-list-number pool name*
6. **interface** *type number*
7. **ip address** *ip-address mask*
8. **ip nat inside**
9. **exit**
10. **interface** *type number*
11. **ip address** *ip-address mask*
12. **ip nat outside**
13. **end**

DETAILED STEPS

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

	Command or Action	Purpose
	Switch# configure terminal	
Step 3	ip nat pool name start-ip end-ip netmask netmask prefix-length prefix-length Example: Switch(config)# ip nat pool net-208 172.16.233.208 172.16.233.223 prefix-length 28	Defines a pool of global addresses to be allocated as needed.
Step 4	access-list access-list-number permit source [source-wildcard] Example: Switch(config)# access-list 1 permit 192.168.34.0 0.0.0.255	Defines a standard access list permitting those addresses that are to be translated.
Step 5	ip nat inside source list access-list-number pool name Example: Switch(config)# ip nat inside source list 1 pool net-208	Establishes dynamic source translation, specifying the access list defined in Step 4.
Step 6	interface type number Example: Switch(config)# interface ethernet 1	Specifies an interface and enters interface configuration mode.
Step 7	ip address ip-address mask Example: Switch(config-if)# ip address 10.114.11.39 255.255.255.0	Sets a primary IP address for the interface.
Step 8	ip nat inside Example: Switch(config-if)# ip nat inside	Connects the interface to the inside network, which is subject to NAT.
Step 9	exit Example: Switch(config-if)#exit	Exits the interface configuration mode and returns to global configuration mode.
Step 10	interface type number Example: Switch(config)# interface ethernet 0	Specifies an interface and enters interface configuration mode.
Step 11	ip address ip-address mask Example: Switch(config-if)# ip address 172.16.232.182 255.255.255.240	Sets a primary IP address for the interface.
Step 12	ip nat outside Example:	Connects the interface to the outside network.

	Command or Action	Purpose
	Switch(config-if)# ip nat outside	
Step 13	end Example: Switch(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring PAT

Perform this task to allow your internal users access to the Internet and conserve addresses in the inside global address pool using overloading of global addresses.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip nat pool** *name start-ip end-ip netmask netmask* | **prefix-length** *prefix-length*
4. **access-list** *access-list-number permit source [source-wildcard]*
5. **ip nat inside source list** *access-list-number pool name overload*
6. **interface** *type number*
7. **ip address** *ip-address mask* [**secondary**]
8. **ip nat inside**
9. **exit**
10. **interface** *type number*
11. **ip address** *ip-address mask* [**secondary**]
12. **ip nat outside**
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip nat pool <i>name start-ip end-ip netmask netmask</i> prefix-length <i>prefix-length</i> Example: Switch(config)# ip nat pool net-208 192.168.202.129 192.168.202.158 netmask 255.255.255.224	Defines a pool of global addresses to be allocated as needed.

	Command or Action	Purpose
Step 4	<p>access-list <i>access-list-number</i> permit <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Switch(config)# access-list 1 permit 192.168.201.30 0.0.0.255</pre>	<p>Defines a standard access list permitting those addresses that are to be translated.</p> <p>The access list must permit only those addresses that are to be translated. (Remember that there is an implicit “deny all” at the end of each access list.) Use of an access list that is too permissive can lead to unpredictable results.</p>
Step 5	<p>ip nat inside source list <i>access-list-number</i> pool <i>name</i> overload</p> <p>Example:</p> <pre>Switch(config)# ip nat inside source list 1 pool net-208 overload</pre>	<p>Establishes dynamic source translation with overloading, specifying the access list defined in Step 4.</p>
Step 6	<p>interface <i>type number</i></p> <p>Example:</p> <pre>Switch(config)# interface ethernet 1</pre>	<p>Specifies an interface and enters interface configuration mode.</p>
Step 7	<p>ip address <i>ip-address mask</i> [secondary]</p> <p>Example:</p> <pre>Switch(config-if)# ip address 192.168.201.1 255.255.255.240</pre>	<p>Sets a primary IP address for an interface.</p>
Step 8	<p>ip nat inside</p> <p>Example:</p> <pre>Switch(config-if)# ip nat inside</pre>	<p>Connects the interface to the inside network, which is subject to NAT.</p>
Step 9	<p>exit</p> <p>Example:</p> <pre>Switch(config-if)# exit</pre>	<p>Exits interface configuration mode and returns to global configuration mode.</p>
Step 10	<p>interface <i>type number</i></p> <p>Example:</p> <pre>Switch(config)# interface ethernet 0</pre>	<p>Specifies a different interface and enters interface configuration mode.</p>
Step 11	<p>ip address <i>ip-address mask</i> [secondary]</p> <p>Example:</p> <pre>Switch(config-if)# ip address 192.168.201.29 255.255.255.240</pre>	<p>Sets a primary IP address for an interface.</p>
Step 12	<p>ip nat outside</p> <p>Example:</p> <pre>Switch(config-if)# ip nat outside</pre>	<p>Connects the interface to the outside network.</p>
Step 13	<p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre>	<p>Exits interface configuration mode and returns to privileged EXEC mode.</p>

Configuring NAT of External IP Addresses Only

By default, NAT translates the addresses embedded in the packet pay-load as explained in [Using Application-Level Gateways with NAT, on page 55](#) section. There might be situations where the translation of the embedded address is not desirable and in such cases, NAT can be configured to translate the external IP address only.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip nat inside source** {list {access-list-number | access-list-name} pool pool-name [overload] | static network local-ip global-ip [no-payload]}
4. **ip nat inside source** {list {access-list-number | access-list-name} pool pool-name [overload] | static {tcp | udp} local-ip local-port global-ip global-port [no-payload]}
5. **ip nat inside source** {list {access-list-number | access-list-name} pool pool-name [overload] | static [network] local-network-mask global-network-mask [no-payload]}
6. **ip nat outside source** {list {access-list-number | access-list-name} pool pool-name | static local-ip global-ip [no-payload]}
7. **ip nat outside source** {list {access-list-number | access-list-name} pool pool-name | static {tcp | udp} local-ip local-port global-ip global-port [no-payload]}
8. **ip nat outside source** {list {access-list-number | access-list-name} pool pool-name | static [network] local-network-mask global-network-mask [no-payload]}
9. **exit**
10. **show ip nat translations** [verbose]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip nat inside source {list {access-list-number access-list-name} pool pool-name [overload] static network local-ip global-ip [no-payload]} Example: Device(config)# ip nat inside source static network 10.1.1.1 192.168.251.0/24 no-payload	Disables the network packet translation on the inside host device.
Step 4	ip nat inside source {list {access-list-number access-list-name} pool pool-name [overload] static {tcp udp} local-ip local-port global-ip global-port [no-payload]}	Disables port packet translation on the inside host device.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config)# ip nat inside source static tcp 10.1.1.1 2000 192.168.1.1 2000 no-payload</pre>	
Step 5	<p>ip nat inside source {list {access-list-number access-list-name} pool pool-name [overload] static [network] local-network-mask global-network-mask [no-payload]}</p> <p>Example:</p> <pre>Device(config)# ip nat inside source static 10.1.1.1 192.168.1.1 no-payload</pre>	Disables packet translation on the inside host device.
Step 6	<p>ip nat outside source {list {access-list-number access-list-name} pool pool-name static local-ip global-ip [no-payload]}</p> <p>Example:</p> <pre>Device(config)# ip nat outside source static 10.1.1.1 192.168.1.1 no-payload</pre>	Disables packet translation on the outside host device.
Step 7	<p>ip nat outside source {list {access-list-number access-list-name} pool pool-name static {tcp udp} local-ip local-port global-ip global-port [no-payload]}</p> <p>Example:</p> <pre>Device(config)# ip nat outside source static tcp 10.1.1.1 20000 192.168.1.1 20000 no-payload</pre>	Disables port packet translation on the outside host device.
Step 8	<p>ip nat outside source {list {access-list-number access-list-name} pool pool-name static [network] local-network-mask global-network-mask [no-payload]}</p> <p>Example:</p> <pre>Device(config)# ip nat outside source static network 10.1.1.1 192.168.251.0/24 no-payload</pre>	Disables network packet translation on the outside host device.
Step 9	<p>exit</p> <p>Example:</p> <pre>Device(config)# exit</pre>	Exits global configuration mode and returns to privileged EXEC mode.
Step 10	<p>show ip nat translations [verbose]</p> <p>Example:</p> <pre>Device# show ip nat translations</pre>	Displays active NAT.

Configuring Translation of Overlapping Networks

Configure static translation of overlapping networks if your IP addresses in the stub network are legitimate IP addresses belonging to another network and you want to communicate with those hosts or routers using static translation.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip nat inside source static** *local-ip global-ip*
4. **ip nat outside source static** *local-ip global-ip*
5. **interface** *type number*
6. **ip address** *ip-address mask*
7. **ip nat inside**
8. **exit**
9. **interface** *type number*
10. **ip address** *ip-address mask*
11. **ip nat outside**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	
Step 3	ip nat inside source static <i>local-ip global-ip</i> Example: Switch(config)# ip nat inside source static 10.1.1.1 203.0.113.2	Establishes static translation between an inside local address and an inside global address.
Step 4	ip nat outside source static <i>local-ip global-ip</i> Example: Switch(config)# ip nat outside source static 172.16.0.3 10.1.1.3	Establishes static translation between an outside local address and an outside global address.
Step 5	interface <i>type number</i> Example: Switch(config)# interface ethernet 1	Specifies an interface and enters interface configuration mode.
Step 6	ip address <i>ip-address mask</i> Example: Switch(config-if)# ip address 10.114.11.39 255.255.255.0	Sets a primary IP address for an interface.
Step 7	ip nat inside Example:	Marks the interface as connected to the inside.

	Command or Action	Purpose
	<code>Switch(config-if)# ip nat inside</code>	
Step 8	exit Example: <code>Switch(config-if)# exit</code>	Exits interface configuration mode and returns to global configuration mode.
Step 9	interface <i>type number</i> Example: <code>Switch(config)# interface ethernet 0</code>	Specifies a different interface and enters interface configuration mode.
Step 10	ip address <i>ip-address mask</i> Example: <code>Switch(config-if)# ip address 172.16.232.182 255.255.255.240</code>	Sets a primary IP address for an interface.
Step 11	ip nat outside Example: <code>Switch(config-if)# ip nat outside</code>	Marks the interface as connected to the outside.
Step 12	end Example: <code>Switch(config-if)# end</code>	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring Address Translation Timeouts

You can configure address translation timeouts based on your NAT configuration.

By default, dynamically created translation entries time-out after a period of inactivity to enable the efficient use of various resources. You can change the default values on timeouts, if necessary. The following are the default time-out configurations associated with major translation types :

- Established TCP sessions: 24 hours
- UDP flow: 5 minutes
- ICMP flow: 1 minute

The default timeout values are adequate to address the timeout requirements in most of the deployment scenarios. However, these values can be adjusted/fine-tuned as appropriate. It is recommended not to configure very small timeout values (less than 60 seconds) as it could result in high CPU usage. Refer the [Best Practices for NAT Configuration, on page 55](#) section for more information.

Based on your configuration, you can change the timeouts described in this section.

- If you need to quickly free your global IP address for a dynamic configuration, configure a shorter timeout than the default timeout, by using the **ip nat translation timeout** command. However, the configured timeout should be longer than the other timeouts configured using commands specified in the following steps.

- If a TCP session is not properly closed by a finish (FIN) packet from both sides or during a reset, change the default TCP timeout by using the **ip nat translation tcp-timeout** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip nat translation** *seconds*
4. **ip nat translation udp-timeout** *seconds*
5. **ip nat translation tcp-timeout** *seconds*
6. **ip nat translation finrst-timeout** *seconds*
7. **ip nat translation icmp-timeout** *seconds*
8. **ip nat translation syn-timeout** *seconds*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	ip nat translation <i>seconds</i> Example: Switch(config)# ip nat translation 300	(Optional) Changes the amount of time after which NAT translations time out. The default timeout is 24 hours, and it applies to the aging time for half-entries.
Step 4	ip nat translation udp-timeout <i>seconds</i> Example: Switch(config)# ip nat translation udp-timeout 300	(Optional) Changes the UDP timeout value.
Step 5	ip nat translation tcp-timeout <i>seconds</i> Example: Switch(config)# ip nat translation tcp-timeout 2500	(Optional) Changes the TCP timeout value. The default is 24 hours.
Step 6	ip nat translation finrst-timeout <i>seconds</i> Example: Switch(config)# ip nat translation finrst-timeout 45	(Optional) Changes the finish and reset timeout value. finrst-timeout—The aging time after a TCP session receives both finish-in (FIN-IN) and finish-out (FIN-OUT) requests or after the reset of a TCP session.
Step 7	ip nat translation icmp-timeout <i>seconds</i> Example:	(Optional) Changes the ICMP timeout value.

	Command or Action	Purpose
	<code>Switch(config)# ip nat translation icmp-timeout 45</code>	
Step 8	ip nat translation syn-timeout <i>seconds</i> Example: <code>Switch(config)# ip nat translation syn-timeout 45</code>	(Optional) Changes the synchronous (SYN) timeout value. The synchronous timeout or the aging time is used only when a SYN request is received on a TCP session. When a synchronous acknowledgment (SYNACK) request is received, the timeout changes to TCP timeout.
Step 9	end Example: <code>Switch(config-if)# end</code>	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring Switch Database Management (SDM) Template

Use SDM templates to configure system resources to optimize support for NAT.

After you set the template and the system reboots, you can use the **show sdm prefer** privileged EXEC command to verify the new template configuration. If you enter the **show sdm prefer** command before you enter the **reload** privileged EXEC command, the **show sdm prefer** command shows the template currently in use and the template that will become active after a reload.

Follow these steps to set the SDM template to maximize NAT usage:

SUMMARY STEPS

1. **configure terminal**
2. **sdm prefer nat**
3. **end**
4. **write memory**
5. **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <code>Switch# configure terminal</code>	Enters global configuration mode.
Step 2	sdm prefer nat Example: <code>Switch(config)# sdm prefer nat</code>	Specifies the SDM template to be used on the switch. This template is available under the network-advantage license.
Step 3	end Example: <code>Switch(config)# end</code>	Returns to the privileged EXEC mode.

	Command or Action	Purpose
Step 4	write memory Example: Switch# write memory	Save the current configuration before reload.
Step 5	reload Example: Switch# reload	Reloads the operating system.

Using Application-Level Gateways with NAT

NAT performs translation services on any TCP/UDP traffic that does not carry source and destination IP addresses in the application data stream. Protocols that do not carry the source and destination IP addresses include HTTP, TFTP, telnet, archie, finger, Network Time Protocol (NTP), Network File System (NFS), remote login (rlogin), remote shell (rsh) protocol, and remote copy (rcp).

NAT Application-Level Gateway (ALG) enables certain applications that carry address/port information in their payloads to function correctly across NAT domains. In addition to the usual translation of address/ports in the packet headers, ALGs take care of translating the address/ports present in the payload and setting up temporary mappings.

Best Practices for NAT Configuration

- In cases where both static and dynamic rules are configured, ensure that the local addresses specified in the rules do not overlap. If such an overlap is possible, then the ACL associated with the dynamic rule should exclude the corresponding addresses used by the static rule. Similarly, there must not be any overlap between the global addresses as this could lead to undesired behavior.
- Do not employ loose filtering such as **permit ip any any** in an ACL associated with NAT rule as this could result in unwanted packets being translated.
- Do not share an address pool across multiple NAT rules.
- Do not define the same inside global address in Static NAT and Dynamic Pool. This action can lead to undesirable results.
- Exercise caution while modifying the default timeout values associated with NAT. Small timeout values could result in high CPU usage.
- Exercise caution while manually clearing the translation entries as this could result in the disruption of application sessions.
- ALG packets traversing a NAT enabled interface will get punted to CPU, regardless of the packets being translated or not. Therefore, it is recommended to use dedicated interface(s) just for NAT traffic. For all other types of traffic that does not require NAT translation, use a different interface(s).

Troubleshooting NAT

This section explains the basic steps to troubleshoot and verify NAT.

- Clearly define what NAT is supposed to achieve.
- Verify that correct translation table exists using the **show ip nat translation** command.
- Verify that timer values are correctly configured using the **show ip nat translation verbose** command.
- Check the ACL values for NAT using the **show ip access-list** command
- Check the overall NAT configuration using the **show ip nat statistics** command.
- Use the **clear ip nat translation** command to clear the NAT translational table entries before the timer expires.
- Use `debug nat ip` and `debug nat ip detailed` commands to debug NAT configuration.

For further information on Troubleshooting NAT refer <http://www.cisco.com/c/en/us/support/docs/ip/network-address-translation-nat/8605-13.html>

Feature Information for Network Address Translation

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 NAT

Feature Name	Releases	Feature Information
Support for NetworkAddress Translation	Cisco IOS XE Everest 16.5.1a	This feature was introduced. Network Address Translation (NAT) enables private IP networks that uses unregistered IP address to connect to the internet. NAT operates on a device, usually connecting two networks together, and translates the private addresses in the internal network into a global routable addresses, before packets are forwarded onto another network.



CHAPTER 4

VRRPv3 Protocol Support

- [VRRPv3 Protocol Support, on page 57](#)

VRRPv3 Protocol Support

Virtual Router Redundancy Protocol (VRRP) enables a group of devices to form a single virtual device to provide redundancy. The LAN clients can then be configured with the virtual device as their default gateway. The virtual device, representing a group of devices, is also known as a VRRP group. The VRRP version 3 (v3) Protocol Support feature provides the capability to support IPv4 and IPv6 addresses while VRRP version 2 (v2) only supports IPv4 addresses. This module explains concepts related to VRRPv3 and describes how to create and customize a VRRP group in a network. Benefits of using VRRPv3 Protocol Support include the following:

- Interoperability in multi-vendor environments.
- VRRPv3 supports usage of IPv4 and IPv6 addresses while VRRPv2 only supports IPv4 addresses
- Improved scalability through the use of VRRS Pathways.



Note In this module, VRRP and VRRPv3 are used interchangeably.

Restrictions for VRRPv3 Protocol Support

- VRRPv3 is not intended as a replacement for existing dynamic protocols. VRRPv3 is designed for use over multi-access, multicast, or broadcast capable Ethernet LANs.
- VRRPv3 is supported on Ethernet, Fast Ethernet, Bridge Group Virtual Interface (BVI), and Gigabit Ethernet interfaces, and on Multiprotocol Label Switching (MPLS) Virtual Private Networks (VPNs), VRF-aware MPLS VPNs and VLANs.
- Because of the forwarding delay that is associated with the initialization of a BVI interface, you must not configure the VRRPv3 advertise timer to a value lesser than the forwarding delay on the BVI interface. If you configure the VRRPv3 advertise timer to a value equal to or greater than the forwarding delay on the BVI interface, the setting prevents a VRRP device on a recently initialized BVI interface from unconditionally taking over the primary role. Use the **bridge forward-time** command to set the forwarding

delay on the BVI interface. Use the **vrrp timers advertise** command to set the VRRP advertisement timer.

- VRRPv3 does not support Stateful Switchover (SSO).
- Full network redundancy can only be achieved if VRRP operates over the same network path as the VRRS Pathway redundant interfaces. For full redundancy, the following restrictions apply:
 - VRRS pathways should not share a different physical interface as the parent VRRP group or be configured on a sub-interface having a different physical interface as the parent VRRP group.
 - VRRS pathways should not be configured on Switch Virtual Interface (SVI) interfaces as long as the associated VLAN does not share the same trunk as the VLAN on which the parent VRRP group is configured.

Information About VRRPv3 Protocol Support

VRRPv3 Benefits

Support for IPv4 and IPv6

VRRPv3 supports IPv4 and IPv6 address families while VRRPv2 only supports IPv4 addresses.



Note When VRRPv3 is in use, VRRPv2 is unavailable. For VRRPv3 to be configurable, the **fhrrp version vrrp v3** command must be used in global configuration mode

Redundancy

VRRP enables you to configure multiple devices as the default gateway device, which reduces the possibility of a single point of failure in a network.

Load Sharing

You can configure VRRP in such a way that traffic to and from LAN clients can be shared by multiple devices, thereby sharing the traffic load more equitably between available devices.

Multiple Virtual Devices

VRRP supports up to 255 virtual devices (VRRP groups) on a device physical interface, subject to restrictions in scaling. Multiple virtual device support enables you to implement redundancy and load sharing in your LAN topology. In scaled environments, VRRS Pathways should be used in combination with VRRP control groups.

Multiple IP Addresses

The virtual device can manage multiple IP addresses, including secondary IP addresses. Therefore, if you have multiple subnets configured on an Ethernet interface, you can configure VRRP on each subnet.



Note To utilize secondary IP addresses in a VRRP group, a primary address must be configured on the same group.

Preemption

The redundancy scheme of VRRP enables you to preempt a virtual device backup that has taken over for a failing virtual primary device with a higher priority virtual device backup that has become available.



Note Preemption of a lower priority primary device is enabled with an optional delay.

Advertisement Protocol

VRRP uses a dedicated Internet Assigned Numbers Authority (IANA) standard multicast address for VRRP advertisements. For IPv4, the multicast address is 224.0.0.18. For IPv6, the multicast address is FF02:0:0:0:0:0:12. This addressing scheme minimizes the number of devices that must service the multicasts and allows test equipment to accurately identify VRRP packets on a segment. The IANA has assigned VRRP the IP protocol number 112.

VRRP Device Priority and Preemption

An important aspect of the VRRP redundancy scheme is VRRP device priority. Priority determines the role that each VRRP device plays and what happens if the virtual primary device fails.

If a VRRP device owns the IP address of the virtual device and the IP address of the physical interface, this device will function as a virtual primary device.

Priority also determines if a VRRP device functions as a virtual device backup and the order of ascendancy to becoming a virtual primary device if the virtual primary device fails. You can configure the priority of each virtual device backup with a value of 1 through 254 using the **priority** command (use the **vrrp address-family** command to enter the VRRP configuration mode and access the **priority** option).

For example, if device A, the virtual primary device in a LAN topology, fails, an election process takes place to determine if virtual device backups B or C should take over. If devices B and C are configured with the priorities of 101 and 100, respectively, device B is elected to become virtual primary device because it has the higher priority. If devices B and C are both configured with the priority of 100, the virtual device backup with the higher IP address is elected to become the virtual primary device.

By default, a preemptive scheme is enabled whereby a higher priority virtual device backup that becomes available takes over from the virtual device backup that was elected to become virtual primary device. You can disable this preemptive scheme using the **no preempt** command (use the **vrrp address-family** command to enter the VRRP configuration mode, and enter the **no preempt** command). If preemption is disabled, the virtual device backup that is elected to become virtual primary device remains the primary until the original virtual primary device recovers and becomes primary again.



Note Preemption of a lower priority primary device is enabled with an optional delay.

VRRP Advertisements

The virtual primary device sends VRRP advertisements to other VRRP devices in the same group. The advertisements communicate the priority and state of the virtual primary device. The VRRP advertisements are encapsulated into either IPv4 or IPv6 packets (based on the VRRP group configuration) and sent to the appropriate multicast address assigned to the VRRP group. For IPv4, the multicast address is 224.0.0.18. For IPv6, the multicast address is FF02:0:0:0:0:0:0:12. The advertisements are sent every second by default and the interval is configurable.

Cisco devices allow you to configure millisecond timers, which is a change from VRRPv2. You need to manually configure the millisecond timer values on both the primary and the backup devices. The primary advertisement value displayed in the **show vrrp** command output on the backup devices is always 1 second because the packets on the backup devices do not accept millisecond values.

You must use millisecond timers where absolutely necessary and with careful consideration and testing. Millisecond values work only under favorable circumstances. The use of the millisecond timer values is compatible with third party vendors, as long as they also support VRRPv3. You can specify a timer value between 100 milliseconds and 40000 milliseconds.

How to Configure VRRPv3 Protocol Support

Creating and Customizing a VRRP Group

To create a VRRP group, perform the following task. Steps 6 to 14 denote customizing options for the group, and they are optional:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **fhrp version vrrp v3**
4. **interface** *type number*
5. **vrrp group-id address-family** {**ipv4** | **ipv6**}
6. **address** *ip-address* [**primary** | **secondary**]
7. **description** *group-description*
8. **match-address**
9. **preempt delay minimum** *seconds*
10. **priority** *priority-level*
11. **timers advertise** *interval*
12. **vrrpv2**
13. **vrrs leader** *vrrs-leader-name*
14. **shutdown**
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	fhrp version vrrp v3 Example: Device(config)# fhrp version vrrp v3	Enables the ability to configure VRRPv3 and VRRS. Note When VRRPv3 is in use, VRRPv2 is unavailable. The command fhrp version vrrp v2 is not supported though it is configurable.
Step 4	interface type number Example: Device(config)# interface GigabitEthernet 0/0/0	Enters interface configuration mode.
Step 5	vrrp group-id address-family {ipv4 ipv6} Example: Device(config-if)# vrrp 3 address-family ipv4	Creates a VRRP group and enters VRRP configuration mode.
Step 6	address ip-address [primary secondary] Example: Device(config-if-vrrp)# address 100.0.1.10 primary	Specifies a primary or secondary address for the VRRP group. Note VRRPv3 for IPv6 requires that a primary virtual link-local IPv6 address is configured to allow the group to operate. After the primary link-local IPv6 address is established on the group, you can add the secondary global addresses.
Step 7	description group-description Example: Device(config-if-vrrp)# description group 3	(Optional) Specifies a description for the VRRP group.
Step 8	match-address Example: Device(config-if-vrrp)# match-address	(Optional) Matches secondary address in the advertisement packet against the configured address. <ul style="list-style-type: none"> • Secondary address matching is enabled by default.
Step 9	preempt delay minimum seconds Example:	(Optional) Enables preemption of lower priority primary device with an optional delay. <ul style="list-style-type: none"> • Preemption is enabled by default.

	Command or Action	Purpose
	<code>Device(config-if-vrrp)# preempt delay minimum 30</code>	
Step 10	<p>priority <i>priority-level</i></p> <p>Example:</p> <pre>Device(config-if-vrrp)# priority 3</pre>	<p>(Optional) Specifies the priority value of the VRRP group.</p> <ul style="list-style-type: none"> The priority of a VRRP group is 100 by default.
Step 11	<p>timers advertise <i>interval</i></p> <p>Example:</p> <pre>Device(config-if-vrrp)# timers advertise 1000</pre>	<p>(Optional) Sets the advertisement timer in milliseconds.</p> <ul style="list-style-type: none"> The advertisement timer is set to 1000 milliseconds by default.
Step 12	<p>vrrpv2</p> <p>Example:</p> <pre>Device(config-if-vrrp)# vrrpv2</pre>	<p>(Optional) Enables support for VRRPv2 configured devices in compatibility mode.</p> <ul style="list-style-type: none"> VRRPv2 is not supported.
Step 13	<p>vrrs leader <i>vrrs-leader-name</i></p> <p>Example:</p> <pre>Device(config-if-vrrp)# vrrs leader leader-1</pre>	<p>(Optional) Specifies a leader's name to be registered with VRRS and to be used by followers.</p> <ul style="list-style-type: none"> A registered VRRS name is unavailable by default.
Step 14	<p>shutdown</p> <p>Example:</p> <pre>Device(config-if-vrrp)# shutdown</pre>	<p>(Optional) Disables VRRP configuration for the VRRP group.</p> <ul style="list-style-type: none"> VRRP configuration is enabled for a VRRP group by default.
Step 15	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Configuring the Delay Period Before FHRP Client Initialization

To configure the delay period before the initialization of all FHRP clients on an interface, perform the following task:

SUMMARY STEPS

- enable**
- configure terminal**
- fhrp version vrrp v3**
- interface** *type number*
- fhrp delay** {[**minimum**] [**reload**] *seconds*}
- end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	fhrp version vrrp v3 Example: Device(config)# fhrp version vrrp v3	Enables the ability to configure VRRPv3 and VRRS. Note When VRRPv3 is in use, VRRPv2 is unavailable.
Step 4	interface type number Example: Device(config)# interface GigabitEthernet 0/0/0	Enters interface configuration mode.
Step 5	fhrp delay {[minimum] [reload] seconds} Example: Device(config-if)# fhrp delay minimum 5	Specifies the delay period for the initialization of FHRP clients after an interface comes up. • The range is 0-3600 seconds.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuration Examples for VRRPv3 Protocol Support

Example: Enabling VRRPv3 on a Device

The following example shows how to enable VRRPv3 on a device:

```
Device> enable
Device# configure terminal
Device(config)# fhrp version vrrp v3
Device(config-if-vrrp)# end
```

Example: Creating and Customizing a VRRP Group

The following example shows how to create and customize a VRRP group:

Example: Configuring the Delay Period Before FHRP Client Initialization

```
Device> enable
Device# configure terminal
Device(config)# fhrp version vrrp v3
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# vrrp 3 address-family ipv4
Device(config-if-vrrp)# address 100.0.1.10 primary
Device(config-if-vrrp)# description group 3
Device(config-if-vrrp)# match-address
Device(config-if-vrrp)# preempt delay minimum 30
Device(config-if-vrrp)# end
```



Note In the above example, the **fhrp version vrrp v3** command is used in the global configuration mode.

Example: Configuring the Delay Period Before FHRP Client Initialization

The following example shows how to configure the delay period before FHRP client initialization :

```
Device> enable
Device# configure terminal
Device(config)# fhrp version vrrp v3
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# fhrp delay minimum 5
Device(config-if-vrrp)# end
```



Note In the above example, a five-second delay period is specified for the initialization of FHRP clients after the interface comes up. You can specify a delay period between 0 and 3600 seconds.

Example: VRRP Status, Configuration, and Statistics Details

The following is a sample output of the status, configuration and statistics details for a VRRP group:

```
Device> enable
Device# show vrrp detail

GigabitEthernet1/0/1 - Group 3 - Address-Family IPv4
Description is "group 3"
State is MASTER
State duration 53.901 secs
Virtual IP address is 100.0.1.10
Virtual MAC address is 0000.5E00.0103
Advertisement interval is 1000 msec
Preemption enabled, delay min 30 secs (0 msec remaining)
Priority is 100
Master Router is 10.21.0.1 (local), priority is 100
Master Advertisement interval is 1000 msec (expires in 832 msec)
Master Down interval is unknown
VRRPv3 Advertisements: sent 61 (errors 0) - rcvd 0
VRRPv2 Advertisements: sent 0 (errors 0) - rcvd 0
Group Discarded Packets: 0
  VRRPv2 incompatibility: 0
  IP Address Owner conflicts: 0
```

```

Invalid address count: 0
IP address configuration mismatch : 0
Invalid Advert Interval: 0
Adverts received in Init state: 0
Invalid group other reason: 0
Group State transition:
Init to master: 0
Init to backup: 1 (Last change Sun Mar 13 19:52:56.874)
Backup to master: 1 (Last change Sun Mar 13 19:53:00.484)
Master to backup: 0
Master to init: 0
Backup to init: 0

Device# exit

```

Additional References

Related Documents

Related Topic	Document Title
FHRP commands	First Hop Redundancy Protocols Command Reference
Configuring VRRPv2	<i>Configuring VRRP</i>
VRRPv3 Commands	For complete syntax and usage information for the commands used in this chapter. <i>Command Reference (Catalyst 9500 Series Switches)</i>

Standards and RFCs

Standard/RFC	Title
RFC5798	<i>Virtual Router Redundancy Protocol</i>

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 VRRPv3 Protocol Support

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 5: Feature Information for VRRPv3 Protocol Support

Feature Name	Releases	Feature Information
VRRPv3 Protocol Support	Cisco IOS XE Everest 16.6.1	VRRP enables a group of devices to form a single virtual device to provide redundancy. The LAN clients can then be configured with the virtual device as their default gateway. The virtual device, representing a group of devices, is also known as a VRRP group. The VRRPv3 Protocol Support feature provides the capability to support IPv4 and IPv6 addresses. This feature was introduced.

Glossary

Virtual IP address owner—The VRRP device that owns the IP address of the virtual device. The owner is the device that has the virtual device address as its physical interface address.

Virtual device—One or more VRRP devices that form a group. The virtual device acts as the default gateway device for LAN clients. The virtual device is also known as a VRRP group.

Virtual device backup—One or more VRRP devices that are available to assume the role of forwarding packets if the virtual primary device fails.

Virtual primary device—The VRRP device that is currently responsible for forwarding packets sent to the IP addresses of the virtual device. Usually, the virtual primary device also functions as the IP address owner.

VRRP device—A device that is running VRRP.



CHAPTER 5

Configuring WCCP

This section provides information about configuring WCCP.

- [Introduction, on page 67](#)
- [Prerequisites for WCCP, on page 67](#)
- [Restrictions for WCCP, on page 67](#)
- [Information About WCCP, on page 69](#)
- [How to Configure WCCP, on page 74](#)
- [Configuration Examples for WCCP, on page 83](#)
- [Feature Information for WCCP, on page 87](#)

Introduction

The Web Cache Communication Protocol (WCCP) is a Cisco-developed content-routing technology that intercepts IP packets and redirects those packets to a destination other than that specified in the IP packet. Typically the packets are redirected from their destination web server on the Internet to a content engine that is local to the client. In some WCCP deployment scenarios, redirection of traffic may also be required from the web server to the client. WCCP enables you to integrate content engines into your network infrastructure.

The tasks in this document assume that you have already configured content engines on your network.

Prerequisites for WCCP

- To use WCCP, IP must be configured on the interface connected to the Internet and another interface must be connected to the content engine.
- The interface connected to the content engine must be a Fast Ethernet or Gigabit Ethernet interface.

Restrictions for WCCP

General

The following limitations apply to Web Cache Communication Protocol Version 2 (WCCPv2):

- WCCP works only with IPv4 networks.

- WCCP bypasses Network Address Translation (NAT) when Cisco Express Forwarding is enabled.
- WCCP does not interoperate with NAT and the zone-based firewall configured together in a network.
- Service groups can comprise up to 32 content engines and 32 switches.
- For switches servicing a multicast cluster, the Time To Live (TTL) value must be set at 15 or fewer.
- All content engines in a cluster must be configured to communicate with all devices servicing the cluster.
- Multicast addresses must be from 224.0.0.0 to 239.255.255.255.
- Up to eight service groups are supported at the same time on the same client interface.
- The Layer 2 rewrite forwarding method is supported, but generic routing encapsulation (GRE) is not.
- Direct Layer 2 connectivity to content engines is required; Layer 3 connectivity of one or more hops away is not supported.
- Ternary content addressable memory (TCAM) friendly mask-based assignment is supported, but the hash bucket-based method is not.
- When TCAM space is exhausted, traffic is not redirected; it is forwarded normally.
- The WCCP version 2 standard allows for support of up to 256 distinct masks. However, a Cisco Catalyst 9000 series switch supports only mask assignment tables with a single mask.
- A content engine configured for mask assignment that tries to join a farm where the selected assignment method is hash remains out of the farm as long as the cache engine assignment method does not match that of the existing farm.

Catalyst 9000 Series Switches Access Control Lists

When WCCP is using the mask assignment, any redirect list is merged with the mask information from the appliance and the resulting merged ACL is passed down to the Catalyst 9000 series switch hardware. Only Permit or Deny ACL entries from the redirect list in which the protocol is IP or exactly matches the service group protocol are merged with the mask information from the appliance.

The following restrictions apply to the redirect-list ACL:

- The ACL must be an IPv4 extended ACL.
- Only individual source or destination port numbers may be specified; port ranges cannot be specified.
- The only valid matching criteria are **dscp** and **tos**.
- The use of **fragments**, **time-range**, or **options** keywords, or any TCP flags is not permitted.
- If the redirect ACL does not meet the restrictions shown, the system will log the following error message:

```
WCCP-3-BADACE: Service <service group>, invalid access-list entry (seq:<sequence>,
reason:<reason>)
```

Information About WCCP

WCCP Overview

WCCP uses Cisco Content Engines (or other content engines running WCCP) to localize traffic patterns in the network, enabling content requests to be fulfilled locally. Traffic localization reduces transmission costs and download time.

WCCP enables Cisco IOS XE platforms to transparently redirect content requests. With transparent redirection, users can fulfill content requests locally without configuring their browsers to use a web proxy. Instead, they can use the target URL to request content, and have their requests automatically redirected to a content engine. The word "transparent" in this case means that the end user does not know that a requested file (such as a web page) came from the content engine instead of from the originally specified server.

A content engine receiving a request attempts to service it from its own local cache. If the requested information is not present, the content engine issues its own request to the originally targeted server to get the required information. A content engine retrieving the requested information forwards it to the requesting client and caches it to fulfill future requests, thus maximizing download performance and substantially reducing transmission costs.

WCCP enables a series of content engines, called a content engine cluster, to provide content to a device or multiple devices. Network administrators can easily scale their content engines to manage heavy traffic loads through these clustering capabilities. Cisco clustering technology enables each cluster member to work in parallel, resulting in linear scalability. Clustering content engines greatly improves the scalability, redundancy, and availability of your caching solution. You can cluster up to 32 content engines to scale to your desired capacity.

WCCP Mask Assignment

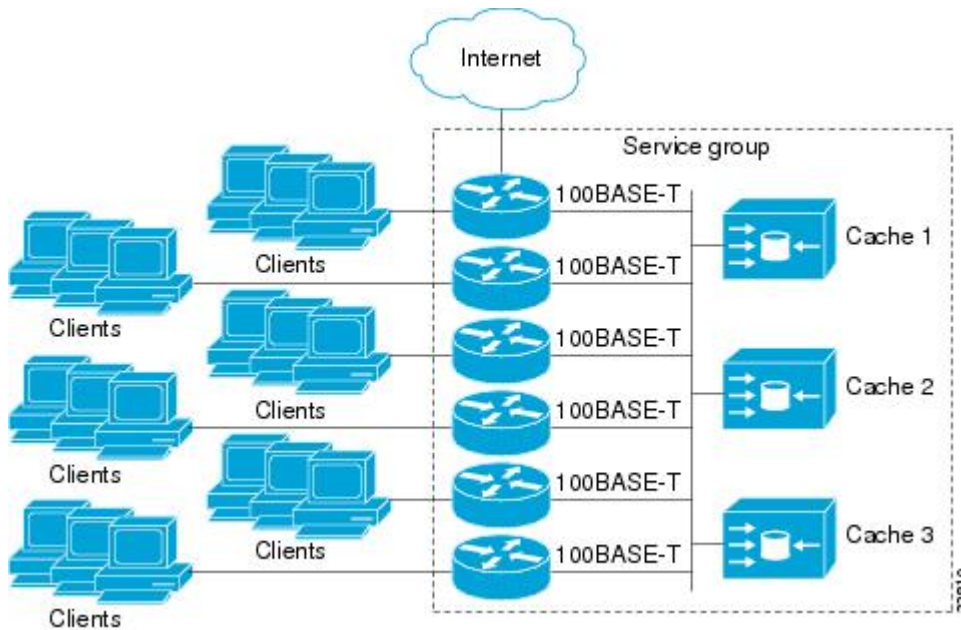
The WCCP Mask Assignment feature enables mask assignment as the load-balancing method (instead of the default hash assignment method) for a WCCP service.

For content engines running Application and Content Networking System (ACNS) software, use the **wccp custom-web-cache** command with the **mask-assign** keyword to configure mask assignment. For content engines running Cisco Wide Area Application Services (WAAS) software, use the **wccp tcp-promiscuous** command with the **mask-assign** keyword to configure mask assignment.

WCCPv2 Configuration

Multiple devices can use WCCPv2 to service a content engine cluster. The figure below illustrates a sample configuration using multiple devices.

Figure 9: Cisco Content Engine Network Configuration Using WCCPv2



The subset of content engines within a cluster and devices connected to the cluster that are running the same service is known as a service group. Available services include TCP and UDP redirection.

WCCPv2 requires that each content engine be aware of all the devices in the service group. To specify the addresses of all the devices in a service group, choose one of the following methods:

- **Unicast**—A list of device addresses for each of the devices in the group is configured on each content engine. In this case, the address of each device in the group must be explicitly specified for each content engine during configuration.
- **Multicast**—A single multicast address is configured on each content engine. In the multicast address method, the content engine sends a single-address notification that provides coverage for all switches in the service group. For example, a content engine could indicate that packets should be sent to a multicast address of 224.0.0.100, which would send a multicast packet to all devices in the service group configured for group listening using WCCP (see the `ip wccp group-listen` interface configuration command for details).

The multicast option is easier to configure because you need only specify a single address on each content engine. This option also allows you to add and remove routers from a service group dynamically, without needing to reconfigure the content engines with a different list of addresses each time.

The following sequence of events details how WCCPv2 configuration works:

1. Each content engine is configured with a list of devices.
2. Each content engine announces its presence and a list of all devices with which it has established communications. The routers reply with their view (list) of content engines in the group.
3. When the view is consistent across all content engines in the cluster, one content engine is designated as the lead and sets the policy that the devices need to deploy in redirecting packets.

WCCPv2 Support for Services Other Than HTTP

WCCPv2 allows redirection of traffic other than HTTP (TCP port 80 traffic), including a variety of UDP and TCP traffic. WCCPv2 supports the redirection of packets intended for other ports, including those used for proxy-web cache handling, File Transfer Protocol (FTP) caching, FTP proxy handling, web caching for ports other than 80, and Real Audio, video, and telephony applications.

To accommodate the various types of services available, WCCPv2 introduced the concept of multiple *service groups*. Service information is specified in the WCCP configuration commands using dynamic services identification numbers (such as 98) or a predefined service keyword (such as **web-cache**). This information is used to validate that service group members are all using or providing the same service.

The content engines in a service group specify traffic to be redirected by protocol (TCP or UDP) and up to eight source or destination ports. Each service group has a priority status assigned to it. The priority of a dynamic service is assigned by the content engine. The priority value is in the range of 0 to 255 where 0 is the lowest priority. The predefined web-cache service has an assigned priority of 240.

WCCPv2 Support for Multiple Devices

WCCPv2 allows multiple devices to be attached to a cluster of cache engines. The use of multiple devices in a service group allows for redundancy, interface aggregation, and distribution of the redirection load. WCCPv2 supports up to 32 devices per service group. Each service group is established and maintained independently.

WCCPv2 MD5 Security

WCCPv2 provides optional authentication that enables you to control which switches and content engines become part of the service group using passwords and the Hashed Message Authentication Code—Message Digest (HMAC MD5) standard. Shared-secret MD5 one-time authentication (set using the **ip wccp password password** global configuration command) enables messages to be protected against interception, inspection, and replay.

WCCPv2 Web Cache Packet Return

If a content engine is unable to provide a requested object it has cached due to error or overload, the content engine will return the request to the device for onward transmission to the originally specified destination server. WCCPv2 provides a check on packets that determines which requests have been returned from the content engine unserved. Using this information, the device can then forward the request to the originally targeted server (rather than attempting to resend the request to the content engine cluster). This process provides error handling transparency to clients.

Typical reasons why a content engine would reject packets and initiate the packet return feature include the following:

- Instances when the content engine is overloaded and has no room to service the packets
- Instances when the content engine is filtering for certain conditions that make caching packets counterproductive (for example, when IP authentication has been turned on)

WCCPv2 Load Distribution

WCCPv2 can be used to adjust the load being offered to individual content engines to provide an effective use of the available resources while helping to ensure high quality of service (QoS) to the clients. WCCPv2 allows the designated content engine to adjust the load on a particular content engine and balance the load across the content engines in a cluster. WCCPv2 uses three techniques to perform load distribution:

- Hot spot handling—Allows an individual hash bucket to be distributed across all the content engines. Prior to WCCPv2, information from one hash bucket could go to only one content engine.
- Load balancing—Allows the set of hash buckets assigned to a content engine to be adjusted so that the load can be shifted from an overwhelmed content engine to other members that have available capacity.
- Load shedding—Enables the switch to selectively redirect the load to avoid exceeding the capacity of a content engine.

The use of these hashing parameters prevents one content engine from being overloaded and reduces the potential for bottlenecks.

WCCP Bypass Packets

WCCP intercepts IP packets and redirects those packets to a destination other than the destination that is specified in the IP header. Typically the packets are redirected from a web server on the Internet to a web cache that is local to the destination.

Occasionally a web cache cannot manage the redirected packets appropriately and returns the packets unchanged to the originating device. These packets are called bypass packets and are returned to the originating device using Layer 2 forwarding without encapsulation (L2). The device decapsulates and forwards the packets normally. The VRF associated with the ingress interface (or the global table if there is no VRF associated) is used to route the packet to the destination.

WCCP Closed Services and Open Services

In applications where packets are intercepted and redirected by a Cisco switch or a router to external WCCP client devices, it may be necessary to block the packets for the application when a WCCP client device is not available. This blocking is achieved by configuring a WCCP closed service. When a WCCP service is configured as closed, the packets that fulfill the services, but do not have an active client device, are discarded.

By default, WCCP operates as an open service, wherein communication between clients and servers proceeds normally in the absence of an intermediary device.

The **ip wccp service-list** command can be used for both closed-mode and open-mode services. Use the **service-list** keyword and *service-access-list* argument to register an application protocol type or port number. Use the **mode** keyword to select an open or closed service.

WCCP Outbound ACL Check

When WCCP is enabled for redirection on an ingress interface, the packets are redirected by WCCP and instead egress on an interface other than the destination that is specified in the IP header. The packets are still subject to ACLs configured on the ingress interface. However, redirection can cause the packets to bypass the ACL configured on the original egress interface. Packets that would have been dropped because of the ACL configured on the original egress interface can be sent out on the redirect egress interface, which poses

a possible security problem. Enabling the WCCP Outbound ACL check feature ensures that redirected packets are subject to any ACL conditions configured on the original egress interface.

WCCP Service Groups

WCCP is a component of Cisco IOS XE software that redirects traffic with defined characteristics from its original destination to an alternative destination. The typical application of WCCP is to redirect traffic bound for a remote web server to a local web cache to improve response time and optimize network resource usage.

The nature of the selected traffic for redirection is defined by service groups (see figure below) specified on content engines and communicated to switches or routers using WCCP.

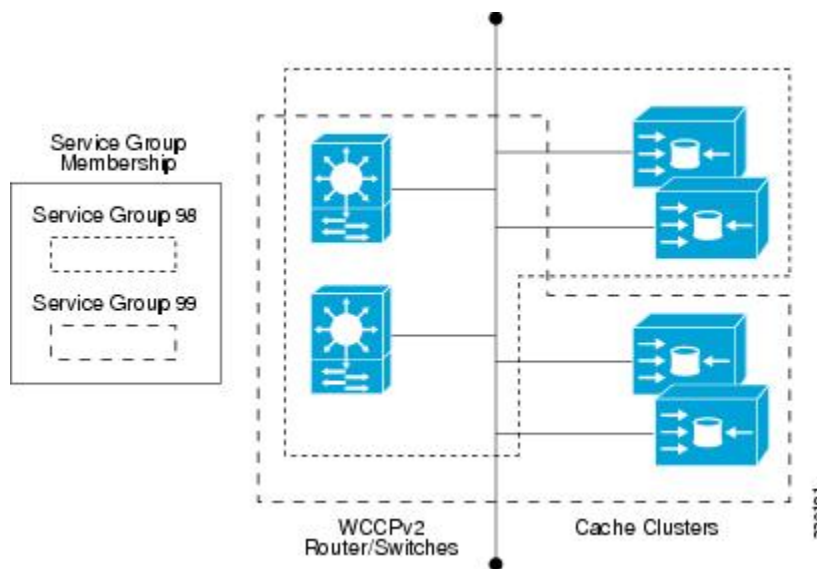
WCCPv2 supports up to 32 switches per service group. Each service group is established and maintained independently.

WCCPv2 uses service groups based on logical redirection services, deployed for intercepting and redirecting traffic. The standard service is web cache, which intercepts TCP port 80 (HTTP) traffic and redirects that traffic to the content engines. This service is referred to as a *well-known service*, because the characteristics of the web cache service are known by both the switch and content engines. A description of a well-known service is not required beyond a service identification. To specify the standard web cache service, use the **ip wccp** command with the **web-cache** keyword.



Note More than one service can run on a switch at the same time, and switches and content engines can be part of multiple service groups at the same time.

Figure 10: WCCP Service Groups



The dynamic services are defined by the content engines; the content engine instructs the switch which protocol or ports to intercept, and how to distribute the traffic. The switch itself does not have information on the characteristics of the dynamic service group’s traffic, because this information is provided by the first content engine to join the group. In a dynamic service, up to eight ports can be specified within a single protocol.

Cisco Content Engines, for example, use dynamic service 99 to specify a reverse-proxy service. However, other content engine devices may use this service number for some other service.

WCCP—Check All Services

An interface may be configured with more than one WCCP service. When more than one WCCP service is configured on an interface, the precedence of a service depends on the relative priority of the service compared to the priority of the other configured services. Each WCCP service has a priority value as part of its definition. When an interface is configured with more than one WCCP service, the precedence of the packets is matched against service groups in priority order.



Note The priority of a WCCP service group cannot be configured via Cisco IOS XE software.

With the **ip wccp check services all** command, WCCP can be configured to check all configured services for a match and perform redirection for those services if appropriate. The caches to which packets are redirected can be controlled by a redirect ACL and by the service priority. The **ip wccp check services all** commands must be configured at global level to support multiple WCCP services.

If no WCCP services are configured with a redirect ACL, the services are considered in priority order until a service is found that matches the IP packet. If no services match the packet, the packet is not redirected. If a service matches the packet and the service has a redirect ACL configured, then the IP packet will be checked against the ACL. If the packet is rejected by the ACL, the packet will not be passed down to lower priority services unless the **ip wccp check services all** command is configured. When the **ip wccp check services all** command is configured, WCCP will continue to attempt to match the packet against any remaining lower priority services configured on the interface.

WCCP Troubleshooting Tips

CPU usage may be very high when WCCP is enabled. The WCCP counters enable a determination of the bypass traffic directly on the switch and can indicate whether the cause is high CPU usage due to enablement of WCCP. In some situations, 10 percent bypass traffic may be normal; in other situations, 10 percent may be high. However, any figure above 25 percent should prompt a closer investigation of what is occurring in the web cache.

If the counters suggest that the level of bypass traffic is high, the next step is to examine the bypass counters in the content engine and determine why the content engine is choosing to bypass the traffic. You can log in to the content engine console and use the CLI to investigate further. The counters allow you to determine the percent of traffic being bypassed.

You can use the **clear wccp** command to remove all WCCP statistics (counts) maintained on the device for a particular service.

You can use the **show wccp** command to display all WCCP global statistics (counts).

How to Configure WCCP

The following configuration tasks assume that you have already installed and configured the content engines you want to include in your network. You must configure the content engines in the cluster before configuring

WCCP functionality on your routers or switches. Refer to the [Cisco Cache Engine User Guide](#) for content engine configuration and setup tasks.

Configuring WCCP

Perform this task to configure WCCP.

Until you configure a WCCP service using the **ip wccp** {**web-cache** | *service-number*} global configuration command, WCCP is disabled on the device. The first use of a form of the **ip wccp** command enables WCCP.

Use the **ip wccp web-cache password** command to set a password for a device and the content engines in a service group. MD5 password security requires that each device and content engine that wants to join a service group be configured with the service group password. The password must be up to eight characters in length. Each content engine or device in the service group will authenticate the security component in a received WCCP packet immediately after validating the WCCP message header. Packets failing authentication will be discarded.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip wccp** {**web-cache** | *service-number*} [**group-address** *multicast-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password* [0 | 7]]
4. **interface** *type number*
5. **ip wccp** {**web-cache** | *service-number*} **redirect** {**in** | **out**}
6. **exit**
7. **interface** *type number*
8. **ip wccp redirect exclude in**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip wccp { web-cache <i>service-number</i> } [group-address <i>multicast-address</i>] [redirect-list <i>access-list</i>] [group-list <i>access-list</i>] [password <i>password</i> [0 7]] Example: Device(config)# ip wccp web-cache password pwd	Specifies a web-cache or dynamic service to enable on a device, specifies the IP multicast address used by the service group, specifies any access lists to use, specifies whether to use MD5 authentication, and enables the WCCP service. <ul style="list-style-type: none"> • Note The password length must not exceed 8 characters.

	Command or Action	Purpose
Step 4	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 0/0	Targets an interface number for which the web cache service will run, and enters interface configuration mode.
Step 5	ip wccp {web-cache <i>service-number</i> } redirect {in out} Example: Device(config-if)# ip wccp web-cache redirect in	Enables packet redirection on an outbound or inbound interface using WCCP. • As indicated by the out and in keyword options, redirection can be specified for outbound interfaces or inbound interfaces.
Step 6	exit Example: Device(config-if)# exit	Exits interface configuration mode.
Step 7	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 0/2/0	Targets an interface number on which to exclude traffic for redirection, and enters interface configuration mode.
Step 8	ip wccp redirect exclude in Example: Device(config-if)# ip wccp redirect exclude in	(Optional) Excludes traffic on the specified interface from redirection.

Configuring Closed Services

Perform this task to specify the number of service groups for WCCP, to configure a service group as a closed or open service, and to optionally specify a check of all services.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Enter one of the following commands:
 - **ip wccp** *service-number* [**service-list** *service-access-list* **mode** {open | closed}]
 - or
 - **ip wccp web-cache mode** {open | closed}
4. **ip wccp check services all**
5. **ip wccp** {web-cache | *service-number*}
6. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>Enter one of the following commands:</p> <ul style="list-style-type: none"> • ip wccp service-number [service-list service-access-list mode {open closed}] • or • ip wccp web-cache mode {open closed} <p>Example:</p> <pre>Device(config)# ip wccp 90 service-list 120 mode closed</pre> <p>or</p> <pre>Device(config)# ip wccp web-cache mode closed</pre>	<p>Configures a dynamic WCCP service as closed or open.</p> <p>or</p> <p>Configures a web-cache service as closed or open.</p> <p>Note When configuring the web-cache service as a closed service, you cannot specify a service access list.</p> <p>Note When configuring a dynamic WCCP service as a closed service, you must specify a service access list.</p>
Step 4	<p>ip wccp check services all</p> <p>Example:</p> <pre>Device(config)# ip wccp check services all</pre>	<p>(Optional) Enables a check of all WCCP services.</p> <ul style="list-style-type: none"> • Use this command to configure WCCP to check the other configured services for a match and perform redirection for those services if appropriate. The caches to which packets are redirected can be controlled by the redirect ACL and not just the service description. <p>Note The ip wccp check services all command is a global WCCP command that applies to all services and is not associated with a single service.</p>
Step 5	<p>ip wccp {web-cache service-number}</p> <p>Example:</p> <pre>Device(config)# ip wccp 201</pre>	<p>Specifies the WCCP service identifier.</p> <ul style="list-style-type: none"> • You can specify the standard web-cache service or a dynamic service number from 0 to 255. • The maximum number of services that can be specified is 256.
Step 6	<p>exit</p> <p>Example:</p>	<p>Exits to privileged EXEC mode.</p>

	Command or Action	Purpose
	Device(config)# exit	

Registering a Device to a Multicast Address

If you decide to use the multicast address option for your service group, you must configure the device to listen for the multicast broadcasts on an interface.

For network configurations where redirected traffic needs to traverse an intervening device, the device being traversed must be configured to perform IP multicast routing. You must configure the following two components to enable traversal over an intervening device:

- Enable IP multicast routing using the **ip multicast-routing** global configuration command.
- Enable the interfaces to which the cache engines will connect to receive multicast transmissions using the **ip wccp group-listen** interface configuration command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast-routing [vrf vrf-name] [distributed]**
4. **ip wccp {web-cache | service-number} group-address multicast-address**
5. **interface type number**
6. **ip pim {sparse-mode | sparse-dense-mode | dense-mode [proxy-register {list access-list | route-map map-name}]}**
7. **ip wccp {web-cache | service-number} group-listen**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip multicast-routing [vrf vrf-name] [distributed] Example: Device(config)# ip multicast-routing	Enables IP multicast routing.
Step 4	ip wccp {web-cache service-number} group-address multicast-address	Specifies the multicast address for the service group.

	Command or Action	Purpose
	Example: Device(config)# ip wccp 99 group-address 239.1.1.1	
Step 5	interface <i>type number</i> Example: Device(config)# interface ethernet 0/0	Enables the interfaces to which the content engines will connect to receive multicast transmissions for which the web cache service will run, and enters interface configuration mode.
Step 6	ip pim { sparse-mode sparse-dense-mode dense-mode [proxy-register { list <i>access-list</i> route-map <i>map-name</i> }] } Example: Device(config-if)# ip pim dense-mode	(Optional) Enables Protocol Independent Multicast (PIM) on an interface. Note To ensure correct operation of the ip wccp group-listen command on Catalyst 9000 series switches, you must enter the ip pim command in addition to the ip wccp group-listen command.
Step 7	ip wccp { web-cache <i>service-number</i> } group-listen Example: Device(config-if)# ip wccp 99 group-listen	Configures an interface to enable or disable the reception of IP multicast packets for WCCP.

Using Access Lists for a WCCP Service Group

Perform this task to configure the device to use an access list to determine which traffic should be directed to which content engines.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	access-list <i>access-list-number</i> remark <i>remark</i> Example: Device(config)# access-list 1 remark Give access to user1	(Optional) Adds a user-friendly comment about an access list entry. <ul style="list-style-type: none"> • A remark of up to 100 characters can precede or follow an access list entry.

	Command or Action	Purpose
Step 4	<p>access-list <i>access-list-number</i> permit {<i>source</i> [<i>source-wildcard</i>] any} [log]</p> <p>Example:</p> <pre>Device(config)# access-list 1 permit 172.16.5.22 0.0.0.0</pre>	<p>Creates an access list that enables or disables traffic redirection to the cache engine and permits the specified source based on a source address and wildcard mask.</p> <ul style="list-style-type: none"> • Every access list needs at least one permit statement; it does not need to be the first entry. • Standard IP access lists are numbered 1 to 99 or 1300 to 1999. • If the <i>source-wildcard</i> is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally use the keyword any as a substitute for the <i>source source-wildcard</i> to specify the source and source wildcard of 0.0.0.0 255.255.255.255. • In this example, host 172.16.5.22 is allowed to pass the access list.
Step 5	<p>access-list <i>access-list-number</i> remark <i>remark</i></p> <p>Example:</p> <pre>Device(config)# access-list 1 remark Give access to user1</pre>	<p>(Optional) Adds a user-friendly comment about an access list entry.</p> <ul style="list-style-type: none"> • A remark of up to 100 characters can precede or follow an access list entry.
Step 6	<p>access-list <i>access-list-number</i> deny {<i>source</i> [<i>source-wildcard</i>] any} [log]</p> <p>Example:</p> <pre>Device(config)# access-list 1 deny 172.16.7.34 0.0.0.0</pre>	<p>Denies the specified source based on a source address and wildcard mask.</p> <ul style="list-style-type: none"> • If the <i>source-wildcard</i> is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally use the abbreviation any as a substitute for the <i>source source-wildcard</i> to specify the source and source wildcard of 0.0.0.0 255.255.255.255. • In this example, host 172.16.7.34 is denied passing the access list.
Step 7	Repeat some combination of Steps 3 through 6 until you have specified the sources on which you want to base your access list.	Remember that all sources not specifically permitted are denied by an implicit deny statement at the end of the access list.
Step 8	<p>ip wccp web-cache group-list <i>access-list</i></p> <p>Example:</p> <pre>Device(config) ip wccp web-cache group-list 1</pre>	Indicates to the device from which IP addresses of content engines to accept packets.
Step 9	<p>ip wccp web-cache redirect-list <i>access-list</i></p> <p>Example:</p>	(Optional) Disables caching for certain clients.

	Command or Action	Purpose
	Device(config)# ip wccp web-cache redirect-list 1	

Enabling the WCCP Outbound ACL Check



Note When all redirection is performed in the hardware, the mode of redirection will change when outbound ACL checking is enabled. The first packet is switched in software to allow the extra ACL check to be performed before a shortcut is installed.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip wccp** {**web-cache** | *service-number*} [**group-address** *multicast-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password*]
4. **ip wccp check acl outbound**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip wccp { web-cache <i>service-number</i> } [group-address <i>multicast-address</i>] [redirect-list <i>access-list</i>] [group-list <i>access-list</i>] [password <i>password</i>] Example: Device(config)# ip wccp web-cache	Enables the support for a Cisco content engine service group or any content engine service group and configures a redirect ACL list or group ACL. Note The web-cache keyword is for WCCP version 1 and version 2 and the <i>service-number</i> argument is for WCCP version 2 only.
Step 4	ip wccp check acl outbound Example: Device(config)# ip wccp check acl outbound	Checks the access control list (ACL) for egress interfaces for packets redirected by WCCP.

	Command or Action	Purpose
Step 5	exit Example: Device(config)# exit	Exits global configuration.

Verifying and Monitoring WCCP Configuration Settings

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show ip wccp [web-cache service-number] [detail view] Example: Device# show ip wccp 24 detail	Displays global information related to WCCP, including the protocol version running, the number of content engines in the router service group, which content engine group is allowed to connect to the router, and which access list is being used. <ul style="list-style-type: none"> • <i>service-number</i>—(Optional) Dynamic number of the web-cache service group being controlled by the content engine. The range is from 0 to 99. For web caches that use Cisco Content Engines, the reverse proxy service is indicated by a value of 99. • web-cache—(Optional) statistics for the web-cache service. • detail—(Optional) other members of a particular service group or web cache that have or have not been detected. • view—(Optional) information about a router or all web caches.
Step 3	show ip interface Example: Device# show ip interface	Displays status about whether any ip wccp redirection commands are configured on an interface; for example, “Web Cache Redirect is enabled / disabled.”
Step 4	more system:running-config Example: Device# more system:running-config	(Optional) Displays contents of the running configuration file (equivalent to the show running-config command).

Configuration Examples for WCCP

Example: Configuring a General WCCPv2 Session

```
Device# configure terminal
Device(config)# ip wccp web-cache group-address 224.1.1.100 password password
Device(config)# ip wccp source-interface GigabitEthernet 0/1/0
Device(config)# ip wccp check services all
! Configures a check of all WCCP services.
Device(config)# interface GigabitEthernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
Device(config-if)# exit
Device(config)# interface GigabitEthernet 0/2/0
Device(config-if)# ip wccp redirect exclude in
Device(config-if)# exit
```

Example: Setting a Password for a Device and Content Engines

```
Device# configure terminal
Device(config)# ip wccp web-cache password password1
```

Example: Configuring a Web Cache Service

```
Device# configure terminal
Device(config)# ip wccp web-cache
Device(config)# interface GigabitEthernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
Device(config-if)# exit
Device# copy running-config startup-config
```

The following example shows how to configure a session in which redirection of HTTP traffic arriving on Gigabit Ethernet interface 0/1/0 is enabled:

```
Device# configure terminal
Device(config)# interface GigabitEthernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
Device(config-if)# exit
Device# show ip interface GigabitEthernet 0/1/0
.
.
.
WCCP Redirect inbound is enabled
WCCP Redirect exclude is disabled
.
.
.
```

Example: Running a Reverse Proxy Service

The following example assumes that you are configuring a service group using Cisco cache engines, which use dynamic service 99 to run a reverse proxy service:

```
Device# configure terminal
Device(config)# ip wccp 99
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp 99 redirect out
```

Example: Registering a Device to a Multicast Address

```
Device# configure terminal
Device(config)# ip wccp web-cache group-address 224.1.1.100
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp web-cache group-listen
```

The following example shows a device configured to run a reverse proxy service, using the multicast address of 224.1.1.1. Redirection applies to packets going out through the Gigabit Ethernet interface 0/1/0:

```
Device# configure terminal
Device(config)# ip wccp 99 group-address 224.1.1.1
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp 99 redirect out
```

Example: Using Access Lists

To achieve better security, you can use a standard access list to notify the device which IP addresses are valid addresses for a content engine attempting to register with the current device. The following example shows a standard access list configuration session where the access list number is 10 for some sample hosts:

```
Device(config)# access-list 10 permit host 10.1.1.1
Device(config)# access-list 10 permit host 10.1.1.2
Device(config)# access-list 10 permit host 10.1.1.3
Device(config)# ip wccp web-cache group-list 10
```

To disable caching for certain clients, servers, or client/server pairs, you can use WCCP access lists. The following example shows that any requests coming from 10.1.1.1 to 10.3.1.1 will bypass the cache, and that all other requests will be serviced normally:

```
Device(config)# ip wccp web-cache redirect-list 120
Device(config)# access-list 120 deny tcp host 10.1.1.1 any
Device(config)# access-list 120 deny tcp any host 10.3.1.1
Device(config)# access-list 120 permit ip any any
```

The following example configures a device to redirect web-related packets received via Gigabit Ethernet interface 0/1/0, destined to any host except 209.165.200.224:

```
Device(config)# access-list 100 deny ip any host 209.165.200.224
Device(config)# access-list 100 permit ip any any
Device(config)# ip wccp web-cache redirect-list 100
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
```

Example: WCCP Outbound ACL Check Configuration

The following configuration example shows that the access list prevents traffic from network 10.0.0.0 leaving Gigabit Ethernet interface 0/1/0. Because the outbound ACL check is enabled, WCCP does not redirect that traffic. WCCP checks packets against the ACL before they are redirected.

```
Device(config)# ip wccp web-cache
Device(config)# ip wccp check acl outbound
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip access-group 10 out
Device(config-if)# exit
Device(config)# ip wccp web-cache redirect-list redirect-out
Device(config)# access-list 10 deny 10.0.0.0 0.255.255.255
Device(config)# access-list 10 permit any
```

If the outbound ACL check is disabled, the HTTP packets from network 10.0.0.0 would be redirected to a web cache. Users with that network address could retrieve web pages even though the network administrator wanted to prevent it.

Example: Verifying WCCP Settings

The following example shows how to verify your configuration changes by using the **more system:running-config** command in privileged EXEC mode. The following example shows that both the web cache service and dynamic service 99 are enabled on the device:

```
Device# more system:running-config

Building configuration...
Current configuration:
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service udp-small-servers
service tcp-small-servers
!
hostname router4
!
enable secret 5 $1$nSVy$faliJsVQXVPW.KuCxZNT1
enable password password1
!
ip subnet-zero
ip wccp web-cache
ip wccp 99
ip domain-name cisco.com
ip name-server 10.1.1.1
ip name-server 10.1.1.2
ip name-server 10.1.1.3
!
!
interface GigabitEthernet0/1/1
ip address 10.3.1.2 255.255.255.0
no ip directed-broadcast
ip wccp web-cache redirect in
ip wccp 99 redirect in
no ip route-cache
no ip mroute-cache
```

```

!
interface GigabitEthernet0/1/0
ip address 10.4.1.1 255.255.255.0
no ip directed-broadcast
ip wccp 99 redirect in
no ip route-cache
no ip mroute-cache
!
interface Serial0
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown
!
interface Serial1
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown
!
ip default-gateway 10.3.1.1
ip classless
ip route 0.0.0.0 0.0.0.0 10.3.1.1
no ip http server
!
!
!
line con 0
transport input none
line aux 0
transport input all
line vty 0 4
password password1
login
!
end

```

The following example shows how to display global statistics related to WCCP:

```
Device# show ip wccp web-cache detail
```

```

WCCP Client information:
WCCP Client ID:      10.1.1.2
Protocol Version:    2.0
State:               Usable
Redirection:        L2
Packet Return:       L2
Packets Redirected:  0
Connect Time:        00:20:34
Assignment:          MASK
Mask  SrcAddr  DstAddr  SrcPort  DstPort
----  -
0000: 0x00000000 0x00001741 0x0000  0x0000
Value SrcAddr  DstAddr  SrcPort  DstPort  CE-IP
-----
0000: 0x00000000 0x00000000 0x0000  0x0000  0x3C010102 (10.1.1.2)
0001: 0x00000000 0x00000001 0x0000  0x0000  0x3C010102 (10.1.1.2)
0002: 0x00000000 0x00000040 0x0000  0x0000  0x3C010102 (10.1.1.2)
0003: 0x00000000 0x00000041 0x0000  0x0000  0x3C010102 (10.1.1.2)
0004: 0x00000000 0x00000100 0x0000  0x0000  0x3C010102 (10.1.1.2)
0005: 0x00000000 0x00000101 0x0000  0x0000  0x3C010102 (10.1.1.2)
0006: 0x00000000 0x00000140 0x0000  0x0000  0x3C010102 (10.1.1.2)

```

For more information about the **show ip wccp web-cache** command, see the *Cisco IOS IP Application Services Command Reference*.

Feature Information for WCCP

Table 6: Feature Information for WCCP

Feature Name	Releases	Feature Information
WCCP Support on Cisco Catalyst 9500 Series Switches	Cisco IOS XE Everest 16.6.1	The Web Cache Communication Protocol (WCCP) is a Cisco-developed content-routing technology that intercepts IP packets and redirects those packets to a destination other than that specified in the IP packet. WCCP enables you to integrate content engines into your network infrastructure.



CHAPTER 6

Configuring Enhanced Object Tracking

- [Finding Feature Information, on page 89](#)
- [Restrictions for Enhanced Object Tracking, on page 89](#)
- [Information About Enhanced Object Tracking, on page 90](#)
- [How to Configure Enhanced Object Tracking, on page 92](#)
- [Monitoring Enhanced Object Tracking, on page 104](#)
- [Feature History for Enhanced Object Tracking, on page 104](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>. An account on Cisco.com is not required.

Restrictions for Enhanced Object Tracking

All tracking configurations must be reconfigured on all Layer 3 subinterfaces after a device reloads. All reload operations on the device must be allowed to complete, so that all Layer 3 subinterfaces are active, before the tracking configuration are reconfigured.



Note This restriction is applicable only for the C9500-32C, C9500-32QC, C9500-48Y4C, and C9500-24Y4C models of the Cisco Catalyst 9500 Series Switches.

Information About Enhanced Object Tracking

Enhanced Object Tracking Overview

Before the introduction of the Enhanced Object Tracking feature, Hot Standby Router Protocol (HSRP) had a simple tracking mechanism that allowed you to track the interface line-protocol state only. If the line-protocol state of the interface went down, the HSRP priority of the router was reduced, allowing another HSRP router with a higher priority to become active.

The Enhanced Object Tracking feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by processes other than HSRP. This feature allows the tracking of other objects in addition to the interface line-protocol state.

A client process such as HSRP, Virtual Router Redundancy Protocol (VRRP), or Gateway Load Balancing Protocol (GLBP), can register its interest in tracking objects and then be notified when the tracked object changes state.

Each tracked object has a unique number that is specified in the tracking command-line interface (CLI). Client processes use this number to track a specific object. The tracking process periodically polls the tracked object for value changes and sends any changes (as up or down values) to interested client processes, either immediately or after a specified delay. Several clients can track the same object, and can take different actions when the object changes state.

You can also track a combination of objects in a list by using either a weight threshold or a percentage threshold to measure the state of the list. You can combine objects using Boolean logic. A tracked list with a Boolean “AND” function requires that each object in the list be in an up state for the tracked object to be up. A tracked list with a Boolean “OR” function needs only one object in the list to be in the up state for the tracked object to be up.

Tracking Interface Line-Protocol or IP Routing State

You can track either the interface line protocol state or the interface IP routing state. When you track the IP routing state, these three conditions are required for the object to be up:

- IP routing must be enabled and active on the interface.
- The interface line-protocol state must be up.
- The interface IP address must be known.

If all three of these conditions are not met, the IP routing state is down.

Tracked Lists

You can configure a tracked list of objects with a Boolean expression, a weight threshold, or a percentage threshold. A tracked list contains one or more objects. An object must exist before it can be added to the tracked list.

- You configure a Boolean expression to specify calculation by using either “AND” or “OR” operators.

- When you measure the tracked list state by a weight threshold, you assign a weight number to each object in the tracked list. The state of the tracked list is determined by whether or not the threshold was met. The state of each object is determined by comparing the total weight of all objects against a threshold weight for each object.
- When you measure the tracked list by a percentage threshold, you assign a percentage threshold to all objects in the tracked list. The state of each object is determined by comparing the assigned percentages of each object to the list.

Tracking Other Characteristics

You can also use the enhanced object tracking for tracking other characteristics.

- You can track the reachability of an IP route by using the **track ip route reachability** global configuration command.
- You can use the **track ip route metric threshold** global configuration command to determine if a route is above or below threshold.
- You can use the **track resolution** global configuration command to change the metric resolution default values for routing protocols.
- You can use the **track timer tracking** configuration command to configure the tracking process to periodically poll tracked objects.

Use the **show track** privileged EXEC command to verify enhanced object tracking configuration.

IP SLAs Object Tracking

Cisco IOS IP Service Level Agreements (IP SLAs) is a network performance measurement and diagnostics tool that uses active monitoring by generating traffic to measure network performance. Cisco IP SLAs operations collects real-time metrics that you can use for network troubleshooting, design, and analysis.

Object tracking of IP SLAs operations allows clients to track the output from IP SLAs objects and use this information to trigger an action. Every IP SLAs operation maintains an SNMP operation return-code value, such as OK or OverThreshold, that can be interpreted by the tracking process. You can track two aspects of IP SLAs operation: state and reachability. For state, if the return code is OK, the track state is up; if the return code is not OK, the track state is down. For reachability, if the return code is OK or OverThreshold, reachability is up; if not OK, reachability is down.

Static Route Object Tracking

Static routing support using enhanced object tracking provides the ability for the device to use ICMP pings to identify when a pre-configured static route or a DHCP route goes down. When tracking is enabled, the system tracks the state of the route and informs the client when that state changes. Static route object tracking uses Cisco IP SLAs to generate ICMP pings to monitor the state of the connection to the primary gateway.

How to Configure Enhanced Object Tracking

Configuring Tracking for Line State Protocol or IP Routing State on an Interface

Follow these steps to track the line-protocol state or IP routing state of an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track *object-number*interface *interface-id*line-protocol**
4. **delay { *object-number*upseconds[downseconds][upseconds]downseconds}**
5. **exit**
6. **track *object-number*interface *interface-id*ip routing**
7. **delay { *object-number*upseconds[downseconds][upseconds]downseconds}**
8. **end**
9. **show track*object-number***

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	track <i>object-number</i>interface <i>interface-id</i>line-protocol Example: Device(config)# track 33 interface gigabitethernet 1/0/1 line-protocol	(Optional) Creates a tracking list to track the line-protocol state of an interface and enter tracking configuration mode. <ul style="list-style-type: none"> • The <i>object-number</i> identifies the tracked object and can be from 1 to 500. • The interface <i>interface-id</i> is the interface being tracked.
Step 4	delay { <i>object-number</i>upseconds[downseconds][upseconds]downseconds}	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 5	exit	Returns to global configuration mode.

	Command or Action	Purpose
Step 6	track <i>object-number</i> interface <i>interface-id</i> ip routing Example: <pre>Device(config)# track 33 interface gigabitethernet 1/0/1 ip routing</pre>	(Optional) Creates a tracking list to track the IP routing state of an interface and enter tracking configuration mode. IP route tracking tracks an IP route in the routing table and the ability of an interface to route IP packets. <ul style="list-style-type: none"> • The object-number identifies the tracked object and can be from 1 to 500. • The interface <i>interface-id</i> is the interface being tracked.
Step 7	delay { <i>object-number</i> upseconds [downseconds][upseconds] downseconds }	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 8	end	Returns to privileged EXEC mode.
Step 9	show track <i>object-number</i>	Verifies that the specified objects are being tracked.

Configuring Tracked Lists

Configuring a Tracked List with a Weight Threshold

To track by weight threshold, configure a tracked list of objects, specify that weight is used as the threshold, and configure a weight for each of its objects. The state of each object is determined by comparing the total weight of all objects that are up against a threshold weight for each object.

You cannot use the Boolean “NOT” operator in a weight threshold list.

Follow these steps to configure a tracked list of objects by using a weight threshold and to configure a weight for each object:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *track-number***list threshold** {**weight**}
4. **object** *object-number*[**weight***weight-number*]
5. **threshold weight** {**upnumber**[[**downnumber**]}]
6. **delay** { **upseconds**[**downseconds**][**upseconds**]**downseconds**}
7. **end**
8. **show track***object-number*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	track track-numberlist threshold {weight} Example: Device(config)# track 4 list threshold weight	Configures a tracked list object, and enters tracking configuration mode. The track-number can be from 1 to 500. <ul style="list-style-type: none"> • threshold—Specifies the state of the tracked list based on a threshold. • weight— Specifies that the threshold is based on weight.
Step 4	object object-number[weightweight-number] Example: Device(config)# object 2 weight 15	Specifies the object to be tracked. The range is from 1 to 500. The optional weightweight-number specifies the threshold weight for the object. The range is from 1 to 255. <p>Note An object must exist before you can add it to a tracked list.</p>
Step 5	threshold weight {upnumber}[downnumber]} Example: Device(config-track)# threshold weight up 30 down 10	(Optional) Specifies the threshold weight. <ul style="list-style-type: none"> • upnumber—The range is from 1 to 255. • downnumber—(Optional)The range depends on the number selected for the upnumber. If you configure the upnumber as 25, the range shown for the down number is 0 to 24.
Step 6	delay {upseconds[downseconds]][upseconds]downseconds}	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 7	end	Returns to privileged EXEC mode.
Step 8	show trackobject-number	Verify that the specified objects are being tracked.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Tracked List with a Percentage Threshold

To track by percentage threshold, configure a tracked list of objects, specify that a percentage will be used as the threshold, and specify a percentage for all objects in the list. The state of the list is determined by comparing the assigned percentage of each object to the list.

You cannot use the Boolean “NOT” operator in a percentage threshold list.

Follow these steps to configure a tracked list of objects by using a percentage threshold:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *track-numberlist* **threshold** {percentage}
4. **object** *object-number*
5. **threshold percentage** {upnumber|[downnumber]}
6. **delay** { upseconds[downseconds]||[upseconds]downseconds }
7. **end**
8. **show track***object-number*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	track <i>track-numberlist</i> threshold {percentage} Example: Device(config)# track 4 list threshold percentage	Configures a tracked list object, and enters tracking configuration mode. The track-number can be from 1 to 500. <ul style="list-style-type: none"> • threshold—Specifies the state of the tracked list based on a threshold. • percentage— Specifies that the threshold is based on percentage.
Step 4	object <i>object-number</i> Example: Device(config)# object 1	Specifies the object to be tracked. The range is from 1 to 500. Note An object must exist before you can add it to a tracked list.

	Command or Action	Purpose
Step 5	threshold percentage { <i>upnumber</i> [[<i>downnumber</i>]]} Example: Device(config)# threshold percentage up 51 down 10	(Optional) Specifies the threshold percentage. <ul style="list-style-type: none"> • upnumber—The range is from 1 to 100. • downnumber—(Optional)The range depends on the number selected for the upnumber. If you configure the upnumber as 25, the range shown for the down number is 0 to 24.
Step 6	delay { upseconds [[downseconds]][[upseconds] downseconds]}	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 7	end	Returns to privileged EXEC mode.
Step 8	show track <i>object-number</i>	Verify that the specified objects are being tracked.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring HSRP Object Tracking

Follow these steps to configure a standby HSRP group to track an object and change the HSRP priority based on the object state:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *object-number* {**interface** *interface-id* {**line-protocol**|**ip routing**}} **ip route** *ip address/prefix-length* {**metric** **threshold**|**reachability**} **list** {**boolean** {**and**|**or**}} | {**threshold** {**weight**|**percentage**}} }
4. **exit**
5. **interface** { *interface-id*
6. **standby**[[*group-number*]] **ip**[[*ip-address*]]**secondary**]]
7. **standby**[[*group-number*]] **track**[[*object-number*]] [**decrement** *priority-decrement*]]
8. **end**
9. **show standby**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device> enable</pre>	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>track <i>object-number</i>{interface <i>interface-id</i>{line-protocol ip routing}} ip route<i>ip address/prefix-length</i>{metric <i>threshold</i>{reachability} list{boolean{and}}} {threshold{weight{percentage}}}}</p>	<p>(Optional) Create a tracking list to track the configured state and enter tracking configuration mode.</p> <ul style="list-style-type: none"> • The object-number identifies the tracked object and can be from 1 to 500. • Enter interface <i>interface-id</i> to select an interface to track. • Enter line-protocol to track the interface line protocol state or enter ip routing to track the interface IP routing state . • Enter ip route<i>ip-address/prefix-length</i> to track the state of an IP route. • Enter metric threshold to track the threshold metric or enter reachability to track if the route is reachable. The default up threshold is 254 and the default down threshold is 255. • Enter list to track objects grouped in a list. <p>Note Repeat this step for each interface to be tracked.</p>
Step 4	exit	Return to global configuration mode.
Step 5	interface { <i>interface-id</i>	Enter interface configuration mode.
Step 6	standby [<i>group-number</i>] ip [<i>ip-address</i> secondary]]	<p>Creates (or enables) the HSRP group by using its number and virtual IP address.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—Enters a group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>—Specifies the virtual IP address of the hot standby router interface. You must enter the

	Command or Action	Purpose
		<p>virtual IP address for at least one of the interfaces; it can be learned on the other interfaces.</p> <ul style="list-style-type: none"> (Optional) secondary—Specifies that the IP address is a secondary hot standby router interface. If this keyword is omitted, the configured address is the primary IP address.
Step 7	standby [<i>group-number</i>] track [<i>object-number</i>] decrement [<i>priority-decrement</i>]	<p>Configures HSRP to track an object and change the hot standby priority based on the state of the object.</p> <ul style="list-style-type: none"> (Optional) <i>group-number</i>—Enters the group number to which the tracking applies. <i>object-number</i>—Enters a number representing the object to be tracked. The range is from 1 to 500; the default is 1. (Optional) secondary—Specifies that the IP address is a secondary hot standby router interface. If this keyword is omitted, the configured address is the primary IP address. (Optional)decrement<i>priority-decrement</i>—Specifies the amount by which the hot standby priority for the router is decremented (or incremented) when the tracked object goes down (or comes back up). The range is from 1 to 255; the default is 10.
Step 8	end	Returns to privileged EXEC mode.
Step 9	show standby	Verifies the standby router IP address and tracking states.
Step 10	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring IP SLAs Object Tracking

Follow these steps to track the state of an IP SLAs operation or the reachability of an IP SLAs IP host:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *object-number* **ip sla** *operation-number* {**state** | **reachability**}
4. **delay** { **upseconds**[**downseconds**][**upseconds**]**downseconds**}
5. **end**

6. `show trackobject-number`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	track object-number ip sla operation-number {state reachability} Example: Device(config)# <code>track 2 ip sla 123 state</code>	Enters tracking configuration mode to track the state of an IP SLAs operation. <ul style="list-style-type: none"> • <i>object-number</i> range is from 1 to 500. • <i>operation-number</i> range is from 1 to 2147483647.
Step 4	delay { upseconds[downseconds][upseconds]downseconds }	(Optional) Specifies a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
Step 5	end	Returns to privileged EXEC mode.
Step 6	show trackobject-number	Verifies that the specified objects are being tracked.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Static Route Object Tracking

Configuring a Primary Interface for Static Routing

Follow these steps to configure a primary interface for static routing:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interfaceinterface-id`

4. `descriptionstring`
5. `ip addressip-address mask[secondary]`
6. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Selects a primary or secondary interface and enters interface configuration mode.
Step 4	description <i>string</i>	Adds a description to the interface.
Step 5	ip address <i>ip-address mask[secondary]</i>	Sets the primary or secondary IP address for the interface.
Step 6	exit	Returns to global configuration mode.

Configuring a Primary Interface for DHCP

Follow these steps to configure a primary interface for DHCP:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface`*interface-id*
4. `description`*string*
5. `ip dhcp client route track`*number*
6. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Selects a primary or secondary interface and enters interface configuration mode.
Step 4	description <i>string</i>	Adds a description to the interface.
Step 5	ip dhcp client route track <i>number</i>	Configures the DHCP client to associate any added routes with the specified track number. Valid numbers are from 1 to 500.
Step 6	exit	Returns to global configuration mode.

Configuring IP SLAs Monitoring Agent

You can configure an IP SLAs agent to ping an IP address using a primary interface and a track object to monitor the state of the agent.

Follow these steps to configure network monitoring with Cisco IP SLAs:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla***operation number*
4. **icmp-echo**{ *destination ip-address|destination hostname*[**source - ipaddr** {*ip-address|hostname***source-interface***interface-id*]
5. **timeout***milliseconds*
6. **frequency***seconds*
7. **threshold***milliseconds*
8. **exit**
9. **ip sla schedule** *operation-number*[**life** {*forever|seconds*}][**start-time***time*][**pending|now|aftertime**][**ageout***seconds*][**recurring**]
10. **track** *object-number***rtr** *operation-number***state***reachability*
11. **end**
12. **show track***object-number*
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip sla <i>operation number</i>	Begins configuring a Cisco IP SLAs operation and enters IP SLA configuration mode.
Step 4	icmp-echo { <i>destination ip-address destination hostname</i> [source - ipaddr { <i>ip-address hostname</i> source-interface <i>interface-id</i>]	Configures a Cisco IP SLAs end-to-end ICMP echo response time operation and enter IP SLAs ICMP echo configuration mode.
Step 5	timeout <i>milliseconds</i>	Sets the amount of time for which the operation waits for a response from its request packet.
Step 6	frequency <i>seconds</i>	Sets the rate at which the operation is sent into the network.
Step 7	threshold <i>milliseconds</i>	Sets the rising threshold (hysteresis) that generates a reaction event and stores history information for the operation.
Step 8	exit	Exits IP SLAs ICMP echo configuration mode.
Step 9	ip sla schedule <i>operation-number</i> [life { <i>forever</i> <i>seconds</i> }] [start-time <i>time</i>] [pending <i>now</i>] [fall-time <i>seconds</i>] [recuring] Example: Device(config)# track 2 200 state	Configures the scheduling parameters for a single IP SLAs operation. <ul style="list-style-type: none"> • <i>object-number</i> range is from 1 to 500. • <i>operation-number</i> range is from 1 to 2147483647.
Step 10	track <i>object-number</i> rtr <i>operation-number</i> statereachability	Tracks the state of a Cisco IOS IP SLAs operation and enter tracking configuration mode.
Step 11	end	Returns to privileged EXEC mode.
Step 12	show track <i>object-number</i>	Verifies that the specified objects are being tracked.
Step 13	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Routing Policy and a Default Route

Follow these steps to configure a routing policy for backup static routing by using object tracking.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list***access-list-number*
4. **route-map***map tag*[**permit**|**deny**][*sequence-number*]
5. **match ip address**{*access-list number*[**permit**|**deny**][*sequence-number*]
6. **set ip next-hop dynamic dhcp**
7. **set interface***interface-id*
8. **exit**
9. **ip local policy route-map***map tag*
10. **ip route***prefix mask*{*ip address*|*interface-id*[*ip address*]}[*distance*][*name*][**permanent**|**track***track-number*][*tag tag*]
11. **end**
12. **show ip route track table**
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	access-list <i>access-list-number</i>	Defines an extended IP access list. Configure any optional characteristics.
Step 4	route-map <i>map tag</i> [permit deny][<i>sequence-number</i>]	Enters route-map configuration mode and define conditions for redistributing routes from one routing protocol to another.
Step 5	match ip address { <i>access-list number</i> [permit deny][<i>sequence-number</i>]	Distribute any routes that have a destination network number address that is permitted by a standard or extended access list or performs policy routing on packets. You can enter multiple numbers or names.
Step 6	set ip next-hop dynamic dhcp	For DHCP networks only. Sets the next hop to the gateway that was most recently learned by the DHCP client.
Step 7	set interface <i>interface-id</i>	For static routing networks only. Indicates where to send output packets that pass a match clause of a route map for policy routing.

	Command or Action	Purpose
Step 8	<code>exit</code>	Returns to global configuration mode.
Step 9	<code>ip local policy route-map map tag</code>	Identifies a route map to use for local policy routing.
Step 10	<code>ip route prefix mask {ip address} [interface-id [ip address]] [distance] [[name]] [permanent] [track track-number] [tag tag]</code>	For static routing networks only. Establishes static routes. Entering track track-number specifies that the static route is installed only if the configured track object is up.
Step 11	<code>end</code>	Returns to privileged EXEC mode.
Step 12	<code>show ip route track table</code>	Displays information about the IP route track table.
Step 13	<code>copy running-config startup-config</code> Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring Enhanced Object Tracking

Use the privileged EXEC or user EXEC commands in the table below, to display enhanced object tracking information.

Table 7: Commands for Displaying Tracking Information

Command	Purpose
<code>show ip route track table</code>	Displays information about the IP route track table.
<code>show track [object-number]</code>	Displays information about the all tracking objects.
<code>show track brief</code>	Displays VTP status and configuration for tracked interfaces.
<code>show track interface [brief]</code>	Displays information about tracked interfaces.
<code>show track ip [object-number] [brief] route</code>	Displays information about tracked IP routes.
<code>show track resolution</code>	Displays the resolution of tracked parameters.
<code>show track timer</code>	Displays tracked polling interval timers.

Feature History for Enhanced Object Tracking

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Enhanced Object Tracking	Enhanced object tracking allows advanced tracking compared to HSRP that allows interface line-protocol state tracking only. Support for this feature was introduced only on the C9500-12Q, C9500-16X, C9500-24Q, C9500-40X models of the Cisco Catalyst 9500 Series Switches.
Cisco IOS XE Fuji 16.8.1a	Enhanced Object Tracking	Support for this feature was introduced only on the C9500-32C, C9500-32QC, C9500-48Y4C, and C9500-24Y4C models of the Cisco Catalyst 9500 Series Switches.

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 7

Configuring TCP MSS Adjustment

- [Restrictions for TCP MSS Adjustment, on page 107](#)
- [Information about TCP MSS Adjustment, on page 107](#)
- [Configuring the MSS Value for Transient TCP SYN Packets, on page 108](#)
- [Configuring the MSS Value for IPv6 Traffic, on page 109](#)
- [Example: Configuring the TCP MSS Adjustment, on page 110](#)
- [Example: Configuring the TCP MSS Adjustment for IPv6 traffic, on page 110](#)
- [Feature History for TCP MSS Adjustment, on page 110](#)

Restrictions for TCP MSS Adjustment

Subinterfaces do not support TCP MSS Adjust.

Information about TCP MSS Adjustment

The Transmission Control Protocol (TCP) Maximum Segment Size (MSS) Adjustment feature enables the configuration of the maximum segment size for transient packets that traverse a router, specifically TCP segments with the SYN bit set. Use the `ip tcp adjust-mss` command in interface configuration mode to specify the MSS value on the intermediate router of the SYN packets to avoid truncation.

When a host (usually a PC) initiates a TCP session with a server, it negotiates the IP segment size by using the MSS option field in the TCP SYN packet. The value of the MSS field is determined by the MTU configuration on the host. The default MSS value for a PC is 1500 bytes.

The PPP over Ethernet (PPPoE) standard supports an MTU of only 1492 bytes. The disparity between the host and PPPoE MTU size can cause the router in between the host and the server to drop 1500-byte packets and terminate TCP sessions over the PPPoE network. Even if the path MTU (which detects the correct MTU across the path) is enabled on the host, sessions may be dropped because system administrators sometimes disable the ICMP error messages that must be relayed from the host in order for path MTU to work.

The `ip tcp adjust-mss` command helps prevent TCP sessions from being dropped by adjusting the MSS value of the TCP SYN packets.

The `ip tcp adjust-mss` command is effective only for TCP connections passing through the router.

In most cases, the optimum value for the `max-segment-size` argument of the `ip tcp adjust-mss` command is 1452 bytes. This value plus the 20-byte IP header, the 20-byte TCP header, and the 8-byte PPPoE header add up to a 1500-byte packet that matches the MTU size for the Ethernet link.



Note TCP MSS adjustment-based traffic is always software switched.

Supported Interfaces

TCP MSS Adjust is supported only on the following interfaces:

- Physical Layer 3 interface
- SVI
- Layer 3 port channel
- Layer 3 GRE tunnel

Configuring the MSS Value for Transient TCP SYN Packets

Before you begin

Perform this task to configure the MSS for transient packets that traverse a router, specifically TCP segments with the SYN bit set.

We recommend that you use the **ip tcp adjust-mss 1452** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip tcp adjust-mss** *max-segment-size*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted
Step 2	configure terminal Example: Device# config terminal	Enters the global configuration mode.
Step 3	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 1/0/0	Configures an interface type and enters interface configuration mode.

	Command or Action	Purpose
Step 4	ip tcp adjust-mss <i>max-segment-size</i> Example: Device(config-if) # ip tcp adjust-mss 1452	Adjusts the MSS value of TCP SYN packets going through a router. The max-segment-size argument is the maximum segment size, in bytes. The range is from 500 to 1460.
Step 5	end Example: Device(config-if) # end	Exits to global configuration mode.

Configuring the MSS Value for IPv6 Traffic

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ipv6 tcp adjust-mss** *max-segment-size*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted
Step 2	configure terminal Example: Device# config terminal	Enters the global configuration mode.
Step 3	interface <i>type number</i> Example: Device(config) # interface GigabitEthernet 1/0/0	Configures an interface type and enters interface configuration mode.
Step 4	ipv6 tcp adjust-mss <i>max-segment-size</i> Example: Device(config-if) # ipv6 tcp adjust-mss 1440	Adjusts the MSS value of TCP DF packets going through a device. The max-segment-size argument is the maximum segment size, in bytes. The range is from 40 to 1440.
Step 5	end Example: Device(config-if) # end	Exits interface configuration mode and returns to privileged EXEC mode.

Example: Configuring the TCP MSS Adjustment

```

Device(config)#vpdn enable
Device(config)#no vpdn logging
Device(config)#vpdn-group 1
Device(config-vpdn)#request-dialin
Device(config-vpdn-req-in)#protocol pppoe
Device(config-vpdn-req-in)#exit
Device(config-vpdn)#exit
Device(config)#interface GigabitEthernet 0/0/0
Device(config-if)#ip address 192.168.100.1.255.255.255.0
Device(config-if)#ip tcp adjust-mss 1452
Device(config-if)#ip nat inside
Device(config-if)#exit

```

Example: Configuring the TCP MSS Adjustment for IPv6 traffic

```

Device>enable
Device#configure terminal
Device(config)#interface GigabitEthernet 0/0/0
Device(config)#ipv6 tcp adjust-mss 1440
Device(config)#end

```

Feature History for TCP MSS Adjustment

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Fuji 16.8.1a	Transmission Control Protocol (TCP) Maximum Segment Size (MSS) Adjustment	The TCP MSS Adjustment feature enables the configuration of the maximum segment size for transient packets that traverse a router, specifically TCP segments with the SYN bitset. This feature helps prevent TCP sessions from being dropped by adjusting the MSS value of the TCP SYN packets.

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 8

Enhanced IPv6 Neighbor Discovery Cache Management

- [Enhanced IPv6 Neighbor Discovery Cache Management](#) , on page 111
- [Customizing the Parameters for IPv6 Neighbor Discovery](#) , on page 112
- [Examples: Customizing Parameters for IPv6 Neighbor Discovery](#), on page 113
- [Additional References](#), on page 113
- [Feature Information for IPv6 Neighbor Discovery](#), on page 113

Enhanced IPv6 Neighbor Discovery Cache Management



Note This feature is not supported on the C9500-32C, C9500-32QC, C9500-48Y4C, and C9500-24Y4C models of the Cisco Catalyst 9500 Series Switches.

Neighbor discovery protocol enforces the neighbor unreachability detection process to detect failing nodes, or devices, and changes to link-layer addresses. Neighbor unreachability detection process maintains the reachability information for all paths between hosts and neighboring nodes, including host-to-host, host-to-device, and device-to-host communication.

The neighbor cache maintains mapping information about the IPv6 link-local or global address to the link-layer address. The neighbor cache also maintains the reachability state of the neighbor using the neighbor unreachability detection process. Neighbors can be in one of the following five possible states:

- **DELAY**—Neighbor is pending re-resolution with a limited flow of traffic to this neighbor.
- **INCOMPLETE**—Address resolution is in progress, and the link-layer address is not yet known.
- **PROBE**—Neighbor re-resolution is in progress with a limited flow of traffic to this neighbor.
- **REACHABLE**—Neighbor detected within the last reachable time interval.
- **STALE**—Neighbor requires re-resolution with a limited flow of traffic to this neighbor.

Use the **ipv6 nd na glean** command to configure the neighbor discovery protocol to glean an entry from an unsolicited neighbor advertisement.

Use the **ipv6 nd nud retry** command to configure the neighbor discovery protocol to maintain a neighbor discovery cache entry for a neighbor during a network disruption.

Use the **ipv6 nd cache expire refresh** command to configure the neighbor discovery protocol maintain a neighbor discovery cache entry even when no traffic flows to the neighbor.

Customizing the Parameters for IPv6 Neighbor Discovery

To customize the parameters for IPv6 neighbor discovery, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface type number Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier. Enters the interface configuration mode.
Step 4	ipv6 nd nud retry base interval max-attempts [final-wait-time] Example: Device(config-if)# ipv6 nd nud retry 1 1000 3	Configures the number of times neighbor unreachability detection resends neighbor solicitations.
Step 5	ipv6 nd cache expire expire-time-in-seconds [refresh] Example: Device(config-if)# ipv6 nd cache expire 7200	Configures the length of time before an IPv6 neighbor discovery cache entry expires.
Step 6	ipv6 nd na glean Example: Device(config-if)# ipv6 nd na glean	Configures the length of time before an IPv6 neighbor discovery cache entry expires.
Step 7	end Example: Device(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.
Step 8	show ipv6 interface Example: Device# show ipv6 interface	(Optional) Displays the usability status of interfaces that are configured for IPv6 along with neighbor discovery cache management.

Examples: Customizing Parameters for IPv6 Neighbor Discovery

The following example shows that IPv6 neighbor advertisement gleaning is enabled and the IPv6 neighbor discovery cache expiry is set to 7200 seconds (2 hours):

```
Device> enable
Device# configure terminal
Device(config)# interface Port-channel 189
Device(config-if)# no ip address
Device(config-if)# ipv6 address 2001:BD8::/64
Device(config-if)# ipv6 nd reachable-time 2700000
Device(config-if)# ipv6 nd na glean
Device(config-if)# ipv6 nd cache expire 7200
Device(config-if)# no ipv6 redirects
Device(config-if)# end
```

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	See the <i>IP Addressing Services</i> section of <i>Command Reference (Catalyst 9500 Series Switches)</i>
For information on IPv6 Neighbor Discovery Inspection	See the <i>Security</i> section of <i>Software Configuration Guide (Catalyst 9500 Switches)</i>

Feature Information for IPv6 Neighbor Discovery

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.

Table 8: Feature Information for IPv6 Neighbor Discovery

Feature Name	Releases	Feature Information
Enhanced IPv6 Neighbor Discovery Cache Management	Cisco IOS XE Everest 16.5.1a	Neighbor discovery protocol enforces neighbor unreachability detection, which can detect failing nodes or routers, and changes to link-layer addresses. This feature was introduced for the Cisco Catalyst 9500 Series Switches.

Feature Name	Releases	Feature Information
Enhanced IPv6 Neighbor Discovery Cache Management	Cisco IOS XE Gibraltar 16.11.1	Neighbor discovery protocol enforces neighbor unreachability detection, which can detect failing nodes or routers, and changes to link-layer addresses. This feature was introduced for the Cisco Catalyst 9500 High Performance Series Switches.