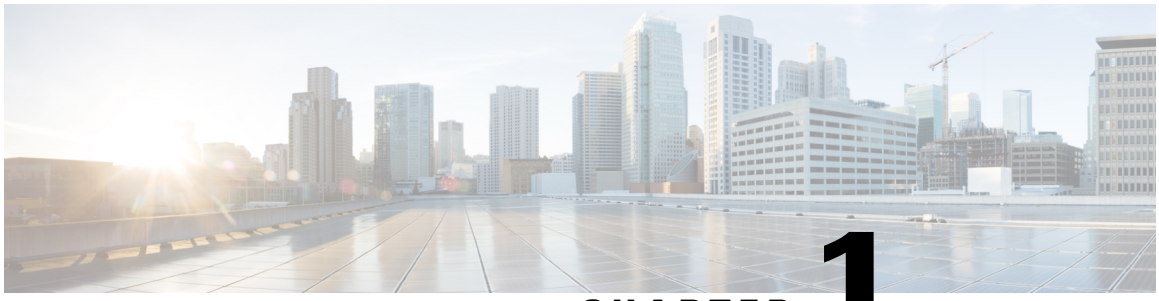
IP Application Services Configuration Guide, Cisco IOS XE Release 3E

First Published: June 30, 2014

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

WCCP Version 2

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.

Cisco IOS Release 12.1 and later releases allow the use of either WCCP Version 1 (WCCPv1) or Version 2 (WCCPv2).

The tasks in this document assume that you have already configured content engines on your network. For specific information on hardware and network planning associated with Cisco Content Engines and WCCP, see the Cisco Content Engines documentation at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/webscale/content/index.htm

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.

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

Prerequisites for WCCP Version 2

- 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.
- Only Catalyst 6500 series switches with a PFC4 support the following hardware capabilities:
  - WCCP generic routing encapsulation (GRE) decapsulation in hardware
  - WCCP egress mask assignment in hardware
  - WCCP exclude capability in hardware

Restrictions for WCCP Version 2

General

The following limitations apply to WCCPv2:

- WCCP works only with IPv4 networks.
- WCCP bypasses Network Address Translation (NAT) when Cisco Express Forwarding is enabled.

WCCPv2

- WCCP works only with IPv4 networks.
- For routers servicing a multicast cluster, the Time To Live (TTL) value must be set at 15 or fewer.
- Service groups can comprise up to 32 content engines and 32 routers.
- All content engines in a cluster must be configured to communicate with all routers servicing the cluster.
- Multicast addresses must be from 224.0.0.0 to 239.255.255.255.

Cisco Catalyst 4500 Series Switches

- Up to eight service groups are supported at the same time on the same client interface.
- The Layer 2 (L2) rewrite forwarding method is supported, but generic routing encapsulation (GRE) is not.
- Direct L2 connectivity to content engines is required; Layer 3 (L3) 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.
- Redirect access control list (ACL) for WCCP on a client interface is not supported.
- Incoming traffic redirection on an interface is supported, but outgoing traffic redirection 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 Catalyst 4500 series switch supports only mask assignment tables with a single mask.

Cisco Catalyst 6500 Series Switches

The following limitation apply to Cisco Catalyst 6500 series switches:

- With a Policy Feature Card 2 (PFC2), Cisco IOS Release 12.2(17d)SXB and later releases support WCCP.
- With a PFC3, Cisco IOS Release 12.2(18)SXD1 and later releases support WCCP.
- With a PFC4, Cisco IOS Release 12.2(50)SY and later releases support WCCP and introduce support for WCCP GRE decapsulation, WCCP mask assignment, and WCCP exclude capability in hardware.
- To use the WCCP Layer 2 PFC redirection feature, configure WCCP on the Catalyst 6500 series switch and configure accelerated WCCP on the cache engine as described in the Transparent Caching document.
- Cisco Application and Content Networking System (ACNS) software releases later than Release 4.2.2 support WCCP Layer 2 Policy Feature Card (PFC) redirection hardware acceleration.
- 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.
- When WCCP Layer 2 PFC redirection is the forwarding method for a service group, the packet counters in the **show ip wccp** *service-number* command output display flow counts instead of packet counts.

Catalyst 6500 Series Switches and Cisco 7600 Series Routers Access Control Lists

When WCCP is using 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 6500 series switch or Cisco 7600 series router 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 simple or extended ACL.
- Only individual source or destination port numbers may be specified; port ranges cannot be specified.
- The only valid matching criteria in addition to individual source or destination port numbers are **dscp** or **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>)
```

WCCP continues to redirect packets, but the redirection is carried out in software (NetFlow Switching) until the access list is adjusted.

Information About WCCP Version 2

WCCPv2 Overview

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

WCCP enables routing platforms to transparently redirect content requests. The main benefit of transparent redirection is that users need not configure 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 router or multiple routers. 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.

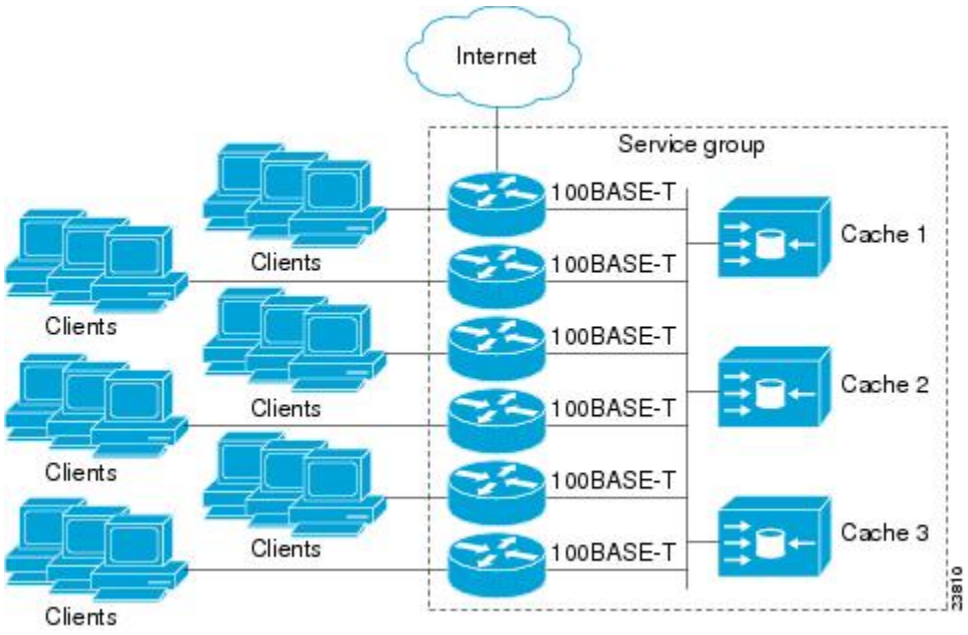
The WCCP Version 2 feature provides several enhancements and features to the WCCP protocol, including:

- The ability of multiple routers to service a content engine cluster.
- Redirection of traffic other than HTTP (TCP port 80 traffic), including a variety of UDP and TCP traffic.
- Optional authentication that enables you to control which routers and content engines become part of the service group using passwords and the HMAC MD5 standard.
- A check on packets that determines which requests have been returned from the content engine unserviced.
- Load adjustments for 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 Configuration

Multiple routers can use WCCPv2 to service a content engine cluster. In WCCPv1, only one router could redirect content requests to a cluster. The figure below illustrates a sample configuration using multiple routers.

Figure 1: Cisco Content Engine Network Configuration Using WCCPv2



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

In WCCPv1, the content engines were configured with the address of the single router. WCCPv2 requires that each content engine be aware of all the routers in the service group. To specify the addresses of all the routers in a service group, you must choose one of the following methods:

- Unicast—A list of router addresses for each of the routers in the group is configured on each content engine. In this case the address of each router 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 routers 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 routers in the service group configured for group listening using WCCP (see the **ip wccp group-listen** or the **ipv6 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 routers.

2. Each content engine announces its presence and a list of all routers 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 routers 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 Routers

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

WCCPv2 MD5 Security

WCCPv2 provides optional authentication that enables you to control which routers 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** [**0** | **7**] *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 router 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 unserviced. Using this information, the router 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 router 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 bottlenecking.

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 router 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 ipv6 wccp***service-id* command to remove the IPv6 WCCP statistics (counts) maintained on the router for a particular service.

You can use the **clear wccp** command to remove all (IPv4 and IPv6) WCCP statistics (counts) maintained on the router for a particular service.

You can use the **show ipv6 wccp** command to display the IPv6 WCCP global statistics (counts).

You can use the **show wccp** command to display all (IPv4 and IPv6) WCCP global statistics (counts).

How to Configure WCCP Version 2

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. By default WCCPv2 is used for services, but you can use WCCPv1 functionality instead. To change the running version of WCCP from Version 2 to Version 1, or to return to WCCPv2 after an initial change, use the **ip wccp version** command in global configuration mode.

If a function is not allowed in WCCPv1, an error prompt will be printed to the screen. For example, if WCCPv1 is running on the device and you try to configure a dynamic service, the following message will be displayed: "WCCP V1 only supports the web-cache service." The **show ip wccp** EXEC command will display the WCCP protocol version number that is running on your device.

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

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip wccp version** {**1** | **2**} **Example:** `Device(config)# ip wccp version 2`	Specifies which version of WCCP to configure on a device. • WCCPv2 is the default running version.
Step 4	**ip wccp** [**vrf** *vrf-name*] {**web-cache** | *service-number*} [**group-address** *multicast-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password* [**0** | **7**]] **Example:** `Device(config)# ip wccp web-cache password pwd`	Specifies a web-cache or dynamic service to enable on a device, specifies a VRF-name to associate with the service group, 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. • **Note** The password length must not exceed 8 characters.
Step 5	**interface** *type number* **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 6	**ip wccp** [**vrf** *vrf-name*] {**web-cache** | *service-number*} **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 7	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode.
Step 8	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/2/0`	Targets an interface number on which to exclude traffic for redirection, and enters interface configuration mode.

	Command or Action	Purpose
Step 9	**ip wccp redirect exclude in** **Example:** `Device(config-if)# ip wccp redirect exclude in`	(Optional) Excludes traffic on the specified interface from redirection.

Verifying and Monitoring WCCP Configuration Settings

SUMMARY STEPS

1. **enable**
2. **show ip wccp** [**vrf** *vrf-name*] [**web-cache** |*service-number*] [**detail view**]
3. **show ip interface**
4. **more system:running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show ip wccp** [**vrf** *vrf-name*] [**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. • **vrf** *vrf-name*—(Optional) Virtual routing and forwarding (VRF) instance associated with a service group. • *service-number*—(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.

	Command or Action	Purpose
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 Version 2

Example: Changing the Version of WCCP on a Router

The following example shows how to change the WCCP version from the default of WCCPv2 to WCCPv1, and enabling the web-cache service in WCCPv1:

```
Device# show ip wccp

% WCCP version 2 is not enabled
Device# configure terminal

Device(config)# ip wccp version 1

Device(config)# end
Device# show ip wccp

% WCCP version 1 is not enabled
Device# configure terminal

Device(config)# ip wccp web-cache
Device(config)# end
Device# show ip wccp

Global WCCP information:
    Router information:
        Router Identifier:                   10.4.9.8
        Protocol Version:                    1.0
.
.
.
```

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
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Quality of Service Solutions Command Reference*
MQC and information about attaching policy maps to interfaces	"Applying QoS Features Using the MQC" module
Additional match criteria that can be used for packet classification	"Classifying Network Traffic" module
Marking network traffic	"Marking Network Traffic" module

Standards

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

MIBs

MIB	MIBs Link
• CISCO-CLASS-BASED-QOS-CAPABILITY-MIB • CISCO-CLASS-BASED-QOS-MIB	To locate and download MIBs for selected platforms, Cisco IOS XE Software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

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

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

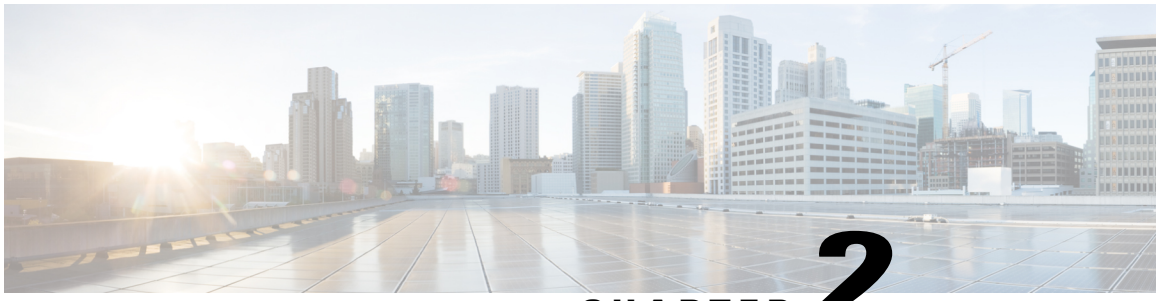
Feature Information for WCCP Version 2

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 . An account on Cisco.com is not required.

Table 1: Feature Information for WCCP Version 2

Feature Name	Releases	Feature Information
WCCP Version 2	Cisco IOS XE Release 3.6E	The WCCP Version 2 feature provides several enhancements and features to the WCCP protocol, including: • The ability of multiple routers to service a content engine cluster. • Redirection of traffic other than HTTP (TCP port 80 traffic), including a variety of UDP and TCP traffic. • Optional authentication that enables you to control which routers and content engines become part of the service group using passwords and the HMAC MD5 standard. • A check on packets that determines which requests have been returned from the content engine unserviced. • Load adjustments for individual content engines to provide an effective use of the available resources while helping to ensure high quality of service (QoS) to the clients. The following commands were introduced or modified by this feature: **clear ip wccp**, **ip wccp**, **ip wccp group-listen**, **ip wccp redirect**, **ip wccp redirect exclude in**, **ip wccp version**, **show ip wccp**. In Cisco IOS XE Release 3.6E, this feature is supported on Cisco Catalyst 3850 Series Switches.



CHAPTER 2

Enhanced Tracking Support

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.

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

Restrictions for Enhanced Object Tracking

Enhanced Object Tracking is not stateful switchover (SSO)-aware and cannot be used with Hot Standby Routing Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), or Gateway Load Balancing Protocol (GLBP) in SSO mode.

Information About Enhanced Tracking Support

Feature Design of Enhanced Object Tracking

The Enhanced Object Tracking feature provides complete separation between the objects to be tracked and the action to be taken by a client when a tracked object changes. Thus, several clients such as HSRP, VRRP, or GLPB can register their interest with the tracking process, track the same object, and each take different action when the object changes.

Each tracked object is identified by a unique number that is specified on the tracking CLI. Client processes use this number to track a specific object.

The tracking process periodically polls the tracked objects and notes any change of value. The changes in the tracked object are communicated to interested client processes, either immediately or after a specified delay. The object values are reported as either up or down.

You can configure a combination of tracked objects in a list and a flexible method for combining objects using Boolean logic. This functionality includes the following capabilities:

- Threshold—The tracked list can be configured to use a weight or percentage threshold to measure the state of the list. Each object in a tracked list can be assigned a threshold weight. The state of the tracked list is determined by whether the threshold has been met.
- Boolean "and" function—When a tracked list has been assigned a Boolean "and" function, each object defined within a subset must be in an up state so that the tracked object can become up.
- Boolean "or" function—When the tracked list has been assigned a Boolean "or" function, at least one object defined within a subset must be in an up state so that the tracked object can become up.

With CSCtg75700, a maximum of 1000 objects can be tracked. Although 1000 tracked objects can be configured, each tracked object uses CPU resources. The amount of available CPU resources on a router depends on variables such as traffic load and how other protocols are configured and run. The ability to use 1000 tracked objects depends on the available CPU. Testing should be conducted on site to ensure that the service works under the specific site traffic conditions.

Interface State Tracking

An IP-routing object is considered up when the following criteria exist:

- IP routing is enabled and active on the interface.
- The interface line-protocol state is up.
- The interface IP address is known. The IP address is configured or received through Dynamic Host Configuration Protocol (DHCP) or IP Control Protocol (IPCP) negotiation.

Interface IP routing will go down when one of the following criteria exists:

- IP routing is disabled globally.
- The interface line-protocol state is down.

- The interface IP address is unknown. The IP address is not configured or received through DHCP or IPCP negotiation.

Tracking the IP-routing state of an interface using the **track interface ip routing** command can be more useful in some situations than just tracking the line-protocol state using the **track interface line-protocol** command, especially on interfaces for which IP addresses are negotiated. For example, on a serial interface that uses the PPP, the line protocol could be up (link control protocol [LCP] negotiated successfully), but IP could be down (IPCP negotiation failed).

The **track interface ip routing** command supports the tracking of an interface with an IP address acquired through any of the following methods:

- Conventional IP address configuration
- PPP/IPCP
- DHCP
- Unnumbered interface

You can configure Enhanced Object Tracking to consider the carrier-delay timer when tracking the IP-routing state of an interface by using the **carrier-delay** command in tracking configuration mode.

Scaled Route Metrics

The **track ip route** command enables tracking of a route in the routing table. If a route exists in the table, the metric value is converted into a number. To provide a common interface to tracking clients, normalize route metric values to the range from 0 to 255, where 0 is connected and 255 is inaccessible. Scaled metrics can be tracked by setting thresholds. Up and down state notification occurs when the thresholds are crossed. The resulting value is compared against threshold values to determine the tracking state as follows:

- State is up if the scaled metric for that route is less than or equal to the up threshold.
- State is down if the scaled metric for that route is greater than or equal to the down threshold.

Tracking uses a per-protocol configurable resolution value to convert the real metric to the scaled metric. The table below shows the default values used for the conversion. You can use the **track resolution** command to change the metric resolution default values.

Table 2: Metric Conversion

Route Type[1]	Metric Resolution
Static	10
Enhanced Interior Gateway Routing Protocol (EIGRP)	2560
Open Shortest Path First (OSPF)	1
Intermediate System-to-Intermediate System (IS-IS)	10

1 RIP is scaled directly to the range from 0 to 255 because its maximum metric is less than 255.

For example, a change in 10 in an IS-IS metric results in a change of 1 in the scaled metric. The default resolutions are designed so that approximately one 2-Mbps link in the path will give a scaled metric of 255.

Scaling the very large metric ranges of EIGRP and IS-IS to a 0 to 255 range is a compromise. The default resolutions will cause the scaled metric to exceed the maximum limit with a 2-Mb/s link. However, this scaling allows a distinction between a route consisting of three Fast-Ethernet links and a route consisting of four Fast-Ethernet links.

Benefits of Enhanced Object Tracking

- Increases the availability and speed of recovery of a network.
- Decreases the number of network outages and their duration.
- Enables client processes such as VRRP and GLBP to track objects individually or as a list of objects. Prior to the introduction of this functionality, the tracking process was embedded within HSRP.

How to Configure Enhanced Tracking Support

Tracking the Line-Protocol State of an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer interface** {*seconds* | **msec** *milliseconds*}
4. **track** *object-number* **interface** *type number* **line-protocol**
5. **carrier-delay**
6. **delay** {**up** *seconds* [**down** [*seconds*] | [**up** *seconds*] **down** *seconds*]}
7. **end**
8. **show track** *object-number*

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**track timer interface** {*seconds* | **msec** *milliseconds*} **Example:** `Device(config)# track timer interface 5`	(Optional) Specifies the interval in which the tracking process polls the tracked object. • The default interval that the tracking process polls interface objects is 1 second. **Note** All polling frequencies can be configured down to 500 milliseconds, overriding the minimum 1-second interval configured using the **msec** keyword and *milliseconds* argument.
Step 4	**track** *object-number* **interface** *type number* **line-protocol** **Example:** `Device(config)# track 3 interface Gigabitethernet 0/0 line-protocol`	Tracks the line-protocol state of an interface and enters tracking configuration mode.
Step 5	**carrier-delay** **Example:** `Device(config-track)# carrier-delay`	(Optional) Enables EOT to consider the carrier-delay timer when tracking the status of an interface.
Step 6	**delay** {**up** *seconds* [**down** [*seconds*] | [**up** *seconds*] **down** *seconds*]} **Example:** `Device(config-track)# delay up 30`	(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.
Step 7	**end** **Example:** `Device(config-track)# end`	Exits to privileged EXEC mode.
Step 8	**show track** *object-number* **Example:** `Device# show track 3`	(Optional) Displays tracking information. • Use this command to verify the configuration.

Example

The following example shows the state of the line protocol on an interface when it is tracked:

```
Device# show track 3

Track 3
   Interface GigabitEthernet 0/0 line-protocol
   Line protocol is Up
     1 change, last change 00:00:05
   Tracked by:
     HSRP GigabitEthernet 0/3 1
```

Tracking the IP-Routing State of an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer interface** {*seconds* | **msec** *milliseconds*}
4. **track** *object-number* **interface** *type number* **ip routing**
5. **carrier-delay**
6. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
7. **end**
8. **show track** *object-number*

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	**track timer interface** {*seconds* | **msec** *milliseconds*} **Example:** `Device(config)# track timer interface 5`	(Optional) Specifies the interval in which the tracking process polls the tracked object. • The default interval that the tracking process polls interface objects is 1 second. **Note** All polling frequencies can be configured down to 500 milliseconds, overriding the minimum 1-second interval configured using the **msec** keyword and *milliseconds* argument.

	Command or Action	Purpose
Step 4	**track** *object-number* **interface** *type number* **ip routing** **Example:** `Device(config)# track 1 interface Gigabitethernet 0/0 ip routing`	Tracks the IP-routing state of an interface and enters tracking configuration mode. • IP-route tracking tracks an IP route in the routing table and the ability of an interface to route IP packets.
Step 5	**carrier-delay** **Example:** `Device(config-track)# carrier-delay`	(Optional) Enables EOT to consider the carrier-delay timer when tracking the status of an interface.
Step 6	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*} **Example:** `Device(config-track)# delay up 30`	(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.
Step 7	**end** **Example:** `Device(config-track)# end`	Returns to privileged EXEC mode.
Step 8	**show track** *object-number* **Example:** `Device# show track 1`	Displays tracking information. • Use this command to verify the configuration.

Example

The following example shows the state of IP routing on an interface when it is tracked:

```
Device# show track 1

Track 1
   Interface GigabitEthernet 0/1 ip routing
   IP routing is Up
     1 change, last change 00:01:08
   Tracked by:
     HSRP GigabitEthernet 0/3 1
```

Tracking IP-Route Reachability

Perform this task to track the reachability of an IP route. A tracked object is considered up when a routing table entry exists for the route and the route is accessible.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer ip route** {*seconds* | **msec** *milliseconds*}
4. **track** *object-number* **ip route** *ip-address*/*prefix-length* **reachability**
5. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
6. **ip vrf** *vrf-name*
7. **end**
8. **show track** *object-number*

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	**track timer ip route** {*seconds* | **msec** *milliseconds*} **Example:** `Device(config)# track timer ip route 20`	(Optional) Specifies the interval in which the tracking process polls the tracked object. • The default interval that the tracking process polls IP-route objects is 15 seconds. **Note** All polling frequencies can be configured down to 500 milliseconds, overriding the minimum 1-second interval configured using the **msec** keyword and *milliseconds* argument.
Step 4	**track** *object-number* **ip route** *ip-address*/*prefix-length* **reachability** **Example:** `Device(config)# track 4 ip route 10.16.0.0/16 reachability`	Tracks the reachability of an IP route and enters tracking configuration mode.

	Command or Action	Purpose
Step 5	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*} **Example:** `Device(config-track)# delay up 30`	(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.
Step 6	**ip vrf** *vrf-name* **Example:** `Device(config-track)# ip vrf VRF2`	(Optional) Configures a VPN routing and forwarding (VRF) table.
Step 7	**end** **Example:** `Device(config-track)# end`	Returns to privileged EXEC mode.
Step 8	**show track** *object-number* **Example:** `Device# show track 4`	(Optional) Displays tracking information. • Use this command to verify the configuration.

Example

The following example shows the state of the reachability of an IP route when it is tracked:

```
Device# show track 4

Track 4
   IP route 10.16.0.0 255.255.0.0 reachability
   Reachability is Up (RIP)
     1 change, last change 00:02:04
   First-hop interface is Ethernet0/1
   Tracked by:
     HSRP Ethernet0/3 1
```

Tracking the Threshold of IP-Route Metrics

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer ip route** {*seconds* | **msec** *milliseconds*}
4. **track resolution ip route** {**eigrp** | **isis** | **ospf** | **static**} *resolution-value*
5. **track** *object-number* **ip route** *ip-address*/*prefix-length* **metric threshold**
6. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
7. **ip vrf** *vrf-name*
8. **threshold metric** {**up** *number* [**down** *number*] | **down** *number* [**up** *number*]}
9. **end**
10. **show track** *object-number*

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	**track timer ip route** {*seconds*	**msec** *milliseconds*} **Example:** `Device(config)# track timer ip route 20`	(Optional) Specifies the interval in which the tracking process polls the tracked object. • The default interval that the tracking process polls IP-route objects is 15 seconds. **Note** All polling frequencies can be configured down to 500 milliseconds, overriding the minimum 1-second interval configured using the **msec** keyword and *milliseconds* argument.
Step 4	**track resolution ip route** {**eigrp**	**isis**	**ospf**	**static**} *resolution-value* **Example:** `Device(config)# track resolution ip route eigrp 300`	(Optional) Specifies resolution parameters for a tracked object. • Use this command to change the default metric resolution values.

	Command or Action	Purpose
Step 5	**track** *object-number* **ip route** *ip-address*/*prefix-length* **metric threshold** **Example:** `Device(config)# track 6 ip route 10.16.0.0/16 metric threshold`	Tracks the scaled metric value of an IP route to determine if it is above or below a threshold and enters tracking configuration mode. • The default down value is 255, which equates to an inaccessible route. • The default up value is 254.
Step 6	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*} **Example:** `Device(config-track)# delay up 30`	(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.
Step 7	**ip vrf** *vrf-name* **Example:** `Device(config-track)# ip vrf VRF1`	(Optional) Configures a VRF table.
Step 8	**threshold metric** {**up** *number* [**down** *number*] | **down** *number* [**up** *number*]} **Example:** `Device(config-track)# threshold metric up 254 down 255`	(Optional) Sets a metric threshold other than the default value.
Step 9	**end** **Example:** `Device(config-track)# end`	Exits to privileged EXEC mode.
Step 10	**show track** *object-number* **Example:** `Device# show track 6`	(Optional) Displays tracking information. • Use this command to verify the configuration.

Example

The following example shows the metric threshold of an IP route when it is tracked:

```
Device# show track 6

Track 6
   IP route 10.16.0.0 255.255.0.0 metric threshold
   Metric threshold is Up (RIP/6/102)
```

```
  1 change, last change 00:00:08
Metric threshold down 255 up 254
First-hop interface is Ethernet0/1
Tracked by:
  HSRP Ethernet0/3 1
```

Configuration Examples for Enhanced Tracking Support

Example: Interface Line Protocol

In the following example, the tracking process is configured to track the line-protocol state of GigabitEthernet interface 1/0/0. HSRP on GigabitEthernet interface 0/0/0 then registers with the tracking process to be informed of any changes to the line-protocol state of GigabitEthernet interface 1/0/0. If the line protocol on GigabitEthernet interface 1/0/0 goes down, the priority of the HSRP group is reduced by 10.

Router A Configuration

```
Device(config)# track 100 interface GigabitEthernet1/0/0 line-protocol
!
Device(config)# interface GigabitEthernet0/0/0
Device(config-if)# ip address 10.1.0.21 255.255.0.0
Device(config-if)# standby 1 preempt
Device(config-if)# standby 1 ip 10.1.0.1
Device(config-if)# standby 1 priority 110
Device(config-if)# standby 1 track 100 decrement 10
```

Router B Configuration

```
Device(config)# track 100 interface GigabitEthernet1/0/0 line-protocol
!
Device(config)# interface GigabitEthernet0/0/0
Device(config-if)# ip address 10.1.0.22 255.255.0.0
Device(config-if)# standby 1 preempt
Device(config-if)# standby 1 ip 10.1.0.1
Device(config-if)# standby 1 priority 105
Device(config-if)# standby 1 track 100 decrement 10
```

Example: Interface IP Routing

In the following example, the tracking process is configured to track the IP-routing capability of GigabitEthernet interface 1/0/0. HSRP on GigabitEthernet interface 0/0/0 then registers with the tracking process to be informed of any changes to the IP-routing state of GigabitEthernet interface 1/0/0. If the IP-routing state on GigabitEthernet interface 1/0/0 goes down, the priority of the HSRP group is reduced by 10.

If both serial interfaces are operational, Router A will be the HSRP active router because it has the higher priority. However, if IP on GigabitEthernet interface 1/0/0 in Router A fails, the HSRP group priority will be reduced and Router B will take over as the active router, thus maintaining a default virtual gateway service to hosts on the 10.1.0.0 subnet.

See the figure below for a sample topology.

Figure 2: Topology for IP-Routing Support

Router A Configuration

```
Device(config)# track 100 interface GigabitEthernet1/0/0 ip routing
!
Device(config)# interface GigabitEthernet0/0/0
Device(config-if)# ip address 10.1.0.21 255.255.0.0
Device(config-if)# standby 1 preempt
Device(config-if)# standby 1 ip 10.1.0.1
Device(config-if)# standby 1 priority 110
Device(config-if)# standby 1 track 100 decrement 10
```

Router B Configuration

```
Device(config)# track 100 interface GigabitEthernet1/0/0 ip routing
!
Device(config)# interface GigabitEthernet0/0/0
Device(config-if)# ip address 10.1.0.22 255.255.0.0
Device(config-if)# standby 1 preempt
Device(config-if)# standby 1 ip 10.1.0.1
Device(config-if)# standby 1 priority 105
Device(config-if)# standby 1 track 100 decrement 10
```

Example: IP-Route Reachability

In the following example, the tracking process is configured to track the reachability of IP route 10.2.2.0/24:

Router A Configuration

```
Device(config)# track 100 ip route 10.2.2.0/24 reachability
!
Device(config)# interface GigabitEthernet0/0/0
Device(config-if)# ip address 10.1.1.21 255.255.255.0
Device(config-if)# standby 1 preempt
Device(config-if)# standby 1 ip 10.1.1.1
Device(config-if)# standby 1 priority 110
Device(config-if)# standby 1 track 100 decrement 10
```

Router B Configuration

```
Device(config)# track 100 ip route 10.2.2.0/24 reachability
!
Device(config)# interface GigabitEthernet0/0/0
Device(config-if)# ip address 10.1.1.22 255.255.255.0
Device(config-if)# standby 1 preempt
```

```
Device(config-if)# standby 1 ip 10.1.1.1
Device(config-if)# standby 1 priority 105
Device(config-if)# standby 1 track 100 decrement 10
```

Example: IP-Route Threshold Metric

In the following example, the tracking process is configured to track the threshold metric of IP route 10.2.2.0/24:

Router A Configuration

```
Device(config)# track 100 ip route 10.2.2.0/24 metric threshold
!
Device(config)# interface GigabitEthernet0/0/0
Device(config-if)# ip address 10.1.1.21 255.255.255.0
Device(config-if)# standby 1 preempt
Device(config-if)# standby 1 ip 10.1.1.1
Device(config-if)# standby 1 priority 110
Device(config-if)# standby 1 track 100 decrement 10
```

Router B Configuration

```
Device(config)# track 100 ip route 10.2.2.0/24 metric threshold
!
Device(config)# interface GigabitEthernet0/0/0
Device(config-if)# ip address 10.1.1.22 255.255.255.0
Device(config-if)# standby 1 preempt
Device(config-if)# standby 1 ip 10.1.1.1
Device(config-if)# standby 1 priority 105
Device(config-if)# standby 1 track 100 decrement 10
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Embedded Event Manager	*Embedded Event Manager Overview*
HSRP concepts and configuration tasks	*Configuring HSRP*
GLBP concepts and configuration tasks	*Configuring GLBP*
IP SLAs commands	*Cisco IOS IP SLAs Command Reference*
VRRP concepts and configuration tasks	*Configuring VRRP*
GLBP, HSRP, and VRRP commands	*Cisco IOS IP Application Services Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

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

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

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

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Embedded Event Manager	*Embedded Event Manager Overview*
HSRP concepts and configuration tasks	*Configuring HSRP*
GLBP concepts and configuration tasks	*Configuring GLBP*
IP SLAs commands	*Cisco IOS IP SLAs Command Reference*
VRRP concepts and configuration tasks	*Configuring VRRP*
GLBP, HSRP, and VRRP commands	*Cisco IOS IP Application Services Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

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

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

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

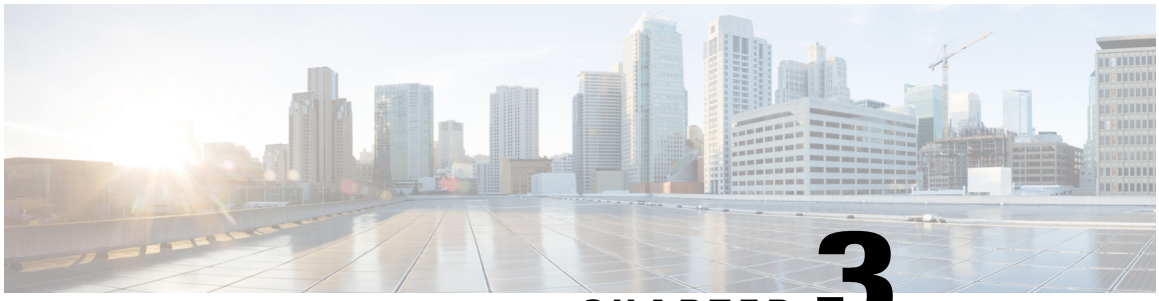
Feature Information for Enhanced Tracking Support

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 . An account on Cisco.com is not required.

Table 3: Feature Information for Enhanced Tracking Support

Feature Name	Releases	Feature Information
Enhanced Tracking Support	Cisco IOS XE Release 3.6E	The Enhanced Tracking Support feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by other Cisco IOS processes as well as HSRP. This feature allows tracking of other objects in addition to the interface line-protocol state. The following commands were introduced or modified by this feature: **debug track** , **delay tracking** , **ip vrf**, **show track** , **standby track** , **threshold metric** , **track interface**, **track ip route**, **track timer**. In Cisco IOS XE Release 3.6E, this feature is supported on Cisco Catalyst 3850 Series Switches.



CHAPTER 3

Object Tracking: IPv6 Route Tracking

The Object Tracking: IPv6 Route Tracking feature expands the Enhanced Object Tracking (EOT) functionality to allow the tracking of IPv6 routes.

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.

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

Restrictions for Object Tracking: IPv6 Route Tracking

Object Tracking: IPv6 Route Tracking is not Stateful Switchover (SSO)-aware and cannot be used with Hot Standby Router Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), or Gateway Load Balancing Protocol (GLBP) in SSO mode.

Information About Object Tracking: IPv6 Route Tracking

Enhanced Object Tracking and IPv6 Route Tracking

Enhanced Object Tracking (EOT) provides complete separation between the objects to be tracked and the action to be taken by a client when a tracked object changes. Thus, several clients such as Hot Standby Router Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), or Gateway Load Balancing Protocol (GLBP) can register interest with a tracking process, track the same object, and each take different a action when the object changes.

Each tracked object is identified by a unique number that is specified on the tracking CLI. Client processes use this number to track a specific object.

A tracking process periodically polls tracked objects and notes any change in value. The changes in the tracked object are communicated to interested client processes, either immediately or after a specified delay. The object values are reported as either up or down.

The Object Tracking: IPv6 Route Tracking feature expands EOT functionality to allow the tracking of IPv6 routes.

How to Configure Object Tracking: IPv6 Route Tracking

Tracking the IPv6-Routing State of an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer interface** {*seconds* | **msec** *milliseconds*}
4. **track** *object-number* **interface** *type number* **ipv6 routing**
5. **carrier-delay**
6. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
7. **end**
8. **show track** *object-number*

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**track timer interface** {*seconds* | **msec** *milliseconds*} **Example:** `Device(config)# track timer interface 5`	(Optional) Specifies the interval that a tracking process polls the tracked interface. • The default interval that the tracking process polls interface objects is 1 second. **Note** All polling frequencies can be configured down to 500 milliseconds, overriding the minimum 1-second interval configured using the **msec** keyword and *milliseconds* argument.
Step 4	**track** *object-number* **interface** *type number* **ipv6 routing** **Example:** `Device(config)# track 1 interface GigabitEthernet 0/0/1 ipv6 routing`	Tracks the IPv6-routing state of an interface and enters tracking configuration mode. • IPv6-route tracking tracks an IPv6 route in the routing table and the ability of an interface to route IPv6 packets.
Step 5	**carrier-delay** **Example:** `Device(config-track)# carrier-delay`	(Optional) Enables enhanced object tracking to consider the carrier-delay timer when tracking the status of an interface.
Step 6	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*} **Example:** `Device(config-track)# delay up 30`	(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object. **Note** The **up** keyword specifies the time to delay the notification of an up event. The **down** keyword specifies the time to delay the notification of a down event.
Step 7	**end** **Example:** `Device(config-track)# end`	Returns to privileged EXEC mode.
Step 8	**show track** *object-number* **Example:** `Device# show track 1`	Displays tracking information. • Use this command to verify the configuration.

Tracking the Threshold of IPv6-Route Metrics

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer ipv6 route** {*seconds* | **msec** *milliseconds*}
4. **track resolution ipv6 route** {**bgp** | **eigrp** | **isis** | **ospf** | **static** } *resolution-value*
5. **track** *object-number* **ipv6 route** *ipv6-address*/*prefix-length* **metric threshold**
6. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
7. **ipv6 vrf** *vrf-name*
8. **threshold metric** {**up** *number* [**down** *number*] | **down** *number* [**up** *number*]}
9. **end**
10. **show track** *object-number*

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	**track timer ipv6 route** {*seconds* | **msec** *milliseconds*} **Example:** `Device(config)# track timer ipv6 route 20`	(Optional) Specifies the interval that a tracking process polls the tracked object. • The default interval that the tracking process polls IPv6-route objects is 15 seconds. **Note** All polling frequencies can be configured down to 500 milliseconds, overriding the minimum 1-second interval configured using the **msec** keyword and *milliseconds* argument.
Step 4	**track resolution ipv6 route** {**bgp** | **eigrp** | **isis** | **ospf** | **static** } *resolution-value* **Example:** `Device(config)# track resolution ipv6 route eigrp 300`	(Optional) Specifies resolution parameters for a tracked object. • Use this command to change the default metric resolution values.

	Command or Action	Purpose
Step 5	**track** *object-number* **ipv6 route** *ipv6-address*/*prefix-length* **metric threshold** **Example:** `Device(config)# track 6 ipv6 route 2001:DB8:0:ABCD::1/10 metric threshold`	Tracks the scaled metric value of an IPv6 route to determine if it is above or below a threshold and enters tracking configuration mode. • The default down value is 255, which equates to an inaccessible route. • The default up value is 254.
Step 6	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*} **Example:** `Device(config-track)# delay up 30`	(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object. **Note** The **up** keyword specifies the time to delay the notification of an up event. The **down** keyword specifies the time to delay the notification of a down event.
Step 7	**ipv6 vrf** *vrf-name* **Example:** `Device(config-track)# ipv6 vrf VRF1`	(Optional) Tracks an IPv6 route in a specific VPN virtual routing and forwarding (VRF) table.
Step 8	**threshold metric** {**up** *number* [**down** *number*] | **down** *number* [**up** *number*]} **Example:** `Device(config-track)# threshold metric up 254 down 255`	(Optional) Sets a metric threshold other than the default value. **Note** The **up** keyword specifies the up threshold. The state is up if the scaled metric for that route is less than or equal to the up threshold. The default up threshold is 254. The **down** keyword specifies the down threshold. The state is down if the scaled metric for that route is greater than or equal to the down threshold. The default down threshold is 255.
Step 9	**end** **Example:** `Device(config-track)# end`	Returns to privileged EXEC mode.
Step 10	**show track** *object-number* **Example:** `Device# show track 6`	(Optional) Displays tracking information. • Use this command to verify the configuration.

Tracking IPv6-Route Reachability

Perform this task to track the reachability of an IPv6 route. A tracked object is considered up when a routing table entry exists for the route and the route is accessible.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer ipv6 route** {*seconds* | **msec** *milliseconds*}
4. **track** *object-number* **ip route** *ip-address*/*prefix-length* **reachability**
5. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
6. **ipv6 vrf** *vrf-name*
7. **end**
8. **show track** *object-number*

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	**track timer ipv6 route** {*seconds*	**msec** *milliseconds*} **Example:** `Device(config)# track timer ipv6 route 20`	(Optional) Specifies the interval that a tracking process polls the tracked object. • The default interval that the tracking process polls IPv6-route objects is 15 seconds. **Note** All polling frequencies can be configured down to 500 milliseconds, overriding the minimum 1-second interval configured using the **msec** keyword and *milliseconds* argument.
Step 4	**track** *object-number* **ip route** *ip-address*/*prefix-length* **reachability** **Example:** `Device(config)# track 4 ipv6 route 2001:DB8:0:AB82::1/10 reachability`	Tracks the reachability of an IPv6 route and enters tracking configuration mode.
Step 5	**delay** {**up** *seconds* [**down** *seconds*]	[**up** *seconds*] **down** *seconds*}	(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.

	Command or Action	Purpose
	Example: `Device(config-track)# delay up 30`	**Note** The **up** keyword specifies the time to delay the notification of an up event. The **down** keyword specifies the time to delay the notification of a down event.
Step 6	**ipv6 vrf** *vrf-name* **Example:** `Device(config-track)# ipv6 vrf VRF2`	(Optional) Configures a VPN virtual routing and forwarding (VRF) table.
Step 7	**end** **Example:** `Device(config-track)# end`	Returns to privileged EXEC mode.
Step 8	**show track** *object-number* **Example:** `Device# show track 4`	(Optional) Displays tracking information. • Use this command to verify the configuration.

Configuration Examples for Object Tracking: IPv6 Route Tracking

Example: Tracking the IPv6-Routing State of an Interface

The following example shows how to configure tracking for IPv6 routing on the GigabitEthernet 0/0/1 interface:

```
Device(config)# track timer interface 5
Device(config)# track 1 interface GigabitEthernet 0/0/1 ipv6 routing
Device(config-track)# carrier-delay
Device(config-track)# delay up 30
Device(config-track)# end
```

Example: Tracking the Threshold of IPv6-Route Metrics

The following example shows how to configure tracking for IPv6 metric thresholds:

```
Device(config)# track timer ipv6 route 20
Device(config)# track resolution ipv6 route eigrp 300
Device(config)# track 6 ipv6 route 2001:DB8:0:ABCD::1/10 metric threshold
Device(config-track)# delay up 30
Device(config-track)# ipv6 vrf VRF1
```

```
Device(config-track)# threshold metric up 254 down 255
Device(config-track)# end
```

Example: Tracking IPv6-Route Reachability

The following example shows how to configure tracking for IPv6-route reachability:

```
Device(config)# track timer ipv6 route 20
Device(config)# track 4 ipv6 route 2001:DB8:0:AB82::1/10 reachability
Device(config-track)# delay up 30
Device(config-track)# ipv6 vrf VRF2
Device(config-track)# end
```

Additional References for Object Tracking: IPv6 Route Tracking

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Object tracking	*Configuring Enhanced Object Tracking*
IP Application Services commands	*Cisco IOS IP Application Services Command Reference*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. 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. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

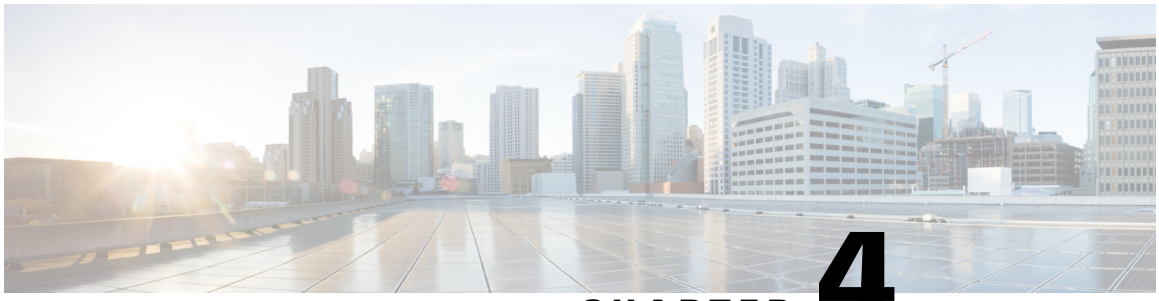
Feature Information for Object Tracking: IPv6 Route Tracking

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 . An account on Cisco.com is not required.

Table 4: Feature Information for Object Tracking: IPv6 Route Tracking

Feature Name	Releases	Feature Information
Object Tracking: IPv6 Route Tracking	Cisco IOS XE Release 3.6E	This feature expands Enhanced Object Tracking (EOT) functionality to allow the tracking of IPv6 routes. In Cisco IOS XE Release 3.6E, this feature is supported on the following platforms: • Cisco Catalyst 3650 Series Switches • Cisco Catalyst 3850 Series Switches • Cisco Catalyst 4500E Supervisor Engine 6-E • Cisco Catalyst 4500E Supervisor Engine 6L-E • Cisco Catalyst 4500E Supervisor Engine 7L-E • Cisco Catalyst 4500E Supervisor Engine 8-E • Cisco Catalyst 4900 Series Switches • Cisco 5700 Series Wireless Controllers



CHAPTER 4

IPv6 Static Route Support for Object Tracking

The IPv6 Static Route Support for Object Tracking feature allows an IPv6 static route to be associated with a tracked-object. A static route is only inserted into the routing information base (RIB) when the tracked object is reachable.

This module provides an overview of the feature and explains how to configure it.

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.

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.

Information About IPv6 Static Route Support for Object Tracking

IPv6 Static Route Support for Object Tracking Overview

Object tracking allows you to track specific objects on a device, such as the interface line protocol state, IP routing, and route reachability, and to take action when the tracked object's state changes. Tracking allows software clients to register interest in the behavior of an object, and receive notifications of changes. This

object represents the state of the system functionality such as the status of an interface (up or down), the existence of an IP prefix in the Routing Information Base (RIB) and so on.

An IPv6 static route creates a tracked object-context for each tracked object. Tracked object contexts are stored in an AVL list that is maintained by the IPv6 static route and indexed by the object number. A tracked-object context is removed from the AVL list when the object is no longer associated with any IPv6 static routes. All IPv6 static routes associated with a tracked object is linked to the tracked object context by an indirect list. An IPv6 static route becomes a client of the tracked objects, and this allows the IPv6 static route to track the state of a tracked object. The **ipv6 route** command allows an IPv6 static route to be associated with a tracked object.

Routing Table Insertion

An IPv6 static route associated with a tracked-object is inserted into the IPv6 routing table if the state of the tracked-object is up and all other routing-table-insertion criteria are met.

The IPv6 Static Route Object Tracking feature uses the IPv6 static deferred state check mechanism to insert or delete a static route into or from the Routing Information Base (RIB). A change in the state of the tracked object is signaled from tracked objects and this causes IPv6 static to insert all IPv6 static routes associated with the tracked object into the state check queue (unless they are already in it). A separate process removes IPv6 static routes from the state check queue and determines whether these routes should be inserted into the RIB or removed from the RIB using the RIB insertion criteria.

Routing Table Insertion Criteria

The following insertion criteria must be met for an IPv6 static route to be inserted into the IPv6 routing table:

1. Interface is up.
2. Next-hop address is not the device's own address.
3. Next-hop address .
4. Next-hop address is resolved.
5. Bidirectional Forwarding Detection (BFD) session is up, if BFD tracking is configured.

Note An IPv6 static route can be associated with a tracked object and a BFD session. Both tracked object and BFD session state must be up before the IPv6 static route is inserted in the routing table.

6. Tracked object state is up.

An IPv6 static route in the routing table is removed if any of the insertion criteria becomes false.

How to Configure IPv6 Static Route Support for Object Tracking

Configuring the IPv6 Static Routing Support for Object Tracking

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 route vrf** *table-name-id ipv6-prefix* {*interface-type interface-number* [*next-hop-ipv6-address*] | *next-hop-ipv6-address*} [*admin-distance* [*multicast-vrf-distance*]] [**multicast**] [**nexthop-vrf** *table-name-id*] [**unicast**] [**tag** *tag-value*] [**track** *object-number*] **name***static-route*]}
4. **end**
5. **show track** *object-number*
6. **show ipv6 static vrf** *id*

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	**ipv6 route vrf** *table-name-id ipv6-prefix* {*interface-type interface-number* [*next-hop-ipv6-address*]	*next-hop-ipv6-address*} [*admin-distance* [*multicast-vrf-distance*]] [**multicast**] [**nexthop-vrf** *table-name-id*] [**unicast**] [**tag** *tag-value*] [**track** *object-number*] **name***static-route*]} **Example:** `Device(config)# ipv6 route vrf 3 2001:DB8:1:2::/64 GigabitEthernet0/0 2001:DB8:3:4::1 track 42`	Establishes static IPv6 routes for all VPN routing and forwarding (VRF) instance tables or a specific VRF table for IPv4 or IPv6 address. • Configure the IPv6 static route object tracking to the static route configuration by using the **track** *object-number* command.
Step 4	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show track** *object-number* **Example:** `Device# show track 42`	Displays information about objects that are tracked by the tracking process.
Step 6	**show ipv6 static vrf** *id* **Example:** `Device(config)# show ipv6 static vrf 3`	Displays static routes that are added to the routing-table, and the reasons if a static route is not added.

The following is sample output from the **show track** command:

```
Device# show track 42

Track 42
  IP route 10.21.12.0 255.255.255.0 reachability
  Reachability is Down (no ip route), delayed Up (1 sec remaining) (connected)
    1 change, last change 00:00:24
  Delay up 20 secs, down 10 secs
  First-hop interface is unknown (was GigabitEthernet1/0)
  Tracked by:
    HSRP GigabitEthernet0/0 3
```

Configuration Examples for IPv6 Static Route Support for Object Tracking

Example: IPv6 Static Route Object Tracking

The following example associates the static route 2001:DB8:1:2::/64 with the state of tracked-object number 42. The static route is inserted in the IPv6 routing table if the state of tracked-object number 42 is up.

```
Device> enable
Device# configure terminal
Device(config)# ipv6 route vrf 3 2001:DB8:1:2::/64 GigabitEthernet0/0 2001:DB8:3:4::1 track
 42
Device(config)# end
```

Additional References for IPv6 Static Route Support for Object Tracking

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
IP Application Services commands	Cisco IOS IP Application Services Command Reference
Object tracking	*Configuring Enhanced Object Tracking*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. 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. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for IPv6 Static Route Support for Object Tracking

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 . An account on Cisco.com is not required.

Table 5: Feature Information for IPv6 Static Route Support for Object Tracking

Feature Name	Releases	Feature Information
IPv6 Static Route Support for Object Tracking	Cisco IOS XE Release 3.6E	This feature expands Enhanced Object Tracking (EOT) functionality to allow the object tracking for IPv6 static routes. In Cisco IOS XE Release 3.6E, this feature is supported on the following platforms: • Catalyst 3650 Series Switches • Catalyst 3850 Series Switches • Catalyst 4500E Supervisor Engine 6-E • Catalyst 4500E Supervisor Engine 7L-E • Catalyst 4500- XE Series Switches • Cisco 5700 Series Wireless Controllers