

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.

- Restrictions for Object Tracking: IPv6 Route Tracking, on page 1
- Information About Object Tracking: IPv6 Route Tracking, on page 1
- How to Configure Object Tracking: IPv6 Route Tracking, on page 2
- Configuration Examples for Object Tracking: IPv6 Route Tracking, on page 6
- Additional References for Object Tracking: IPv6 Route Tracking, on page 7
- Feature Information for Object Tracking: IPv6 Route Tracking, on page 7

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

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**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 do interval configured using the **msec** keyword
Step 2	**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 3	**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 4	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*} **Example:**	(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.

	Command or Action	Purpose
	`Device(config-track)# delay up 30`	**Note** The **up** keyword specifies the time to del time to delay the notification of a down
Step 5	**end** **Example:** `Device(config-track)# end`	Returns to privileged EXEC mode.
Step 6	**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. **track timer ipv6 route** {*seconds* | **msec** *milliseconds*}
2. **track resolution ipv6 route** {**bgp** | **eigrp** | **isis** | **ospf** | **static** } *resolution-value*
3. **track** *object-number* **ipv6 route** *ipv6-address/prefix-length* **metric threshold**
4. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
5. **ipv6 vrf** *vrf-name*
6. **threshold metric** {**up** *number* [**down** *number*] | **down** *number* [**up** *number*]}
7. **end**
8. **show track** *object-number*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**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 configure interval configured using the **msec** keyw

	Command or Action	Purpose
Step 2	**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.
Step 3	**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 4	**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 t time to delay the notification of a down ever
Step 5	**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 6	**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. equal to the up threshold. The default up thr The state is down if the scaled metric for that down threshold is 255.
Step 7	**end** **Example:** `Device(config-track)# end`	Returns to privileged EXEC mode.
Step 8	**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. **track timer ipv6 route** {*seconds* | **msec** *milliseconds*}
2. **track** *object-number* **ip route** *ip-address*/*prefix-length* **reachability**
3. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
4. **ipv6 vrf** *vrf-name*
5. **end**
6. **show track** *object-number*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**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 configure interval configured using the **msec** keyw
Step 2	**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 3	**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 del time to delay the notification of a down
Step 4	**ipv6 vrf** *vrf-name* **Example:**	(Optional) Configures a VPN virtual routing and forwarding (VRF) table.

	Command or Action	Purpose
	`Device(config-track)# ipv6 vrf VRF2`	
Step 5	**end** **Example:** `Device(config-track)# end`	Returns to privileged EXEC mode.
Step 6	**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 www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Object Tracking: IPv6 Route Tracking

Feature Name	Releases	Feature Information
Object Tracking: IPv6 Route Tracking		This feature expands Enhanced Object Tracking (EOT) functionality to allow the tracking of IPv6 routes.