Configuring Object Tracking

This chapter contains the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your 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 "New and Changed Information"chapter or the Feature History table in this chapter.

Information About Object Tracking

Object tracking allows you to track specific objects on the device, such as the interface line protocol state, IP routing, and route reachability, and to take action when the state of the tracked object changes. This feature allows you to increase the availability of the network and shorten recovery time if an object state goes down.

The object tracking feature allows you to create a tracked object that multiple clients can use to modify the client behavior when a tracked object changes. Several clients register their interest with the tracking process, track the same object, and take different actions when the object state changes.

Clients include the following features:

- Embedded Event Manager (EEM)

- Gateway Load Balancing Protocol (GLBP)

- Hot Standby Redundancy Protocol (HSRP)

- Virtual port channel (vPC)

- Virtual Router Redundancy Protocol (VRRP)

The object tracking monitors the status of the tracked objects and communicates any changes made to interested clients. Each tracked object is identified by a unique number that clients can use to configure the action to take when a tracked object changes state.

Cisco NX-OS tracks the following object types:

- Interface line protocol state—Tracks whether the line protocol state is up or down.

- Interface IP routing state—Tracks whether the interface has an IPv4 or IPv6 address and if IPv4 or IPv6 routing is enabled and active.

- IP route reachability—Tracks whether an IPv4 or IPv6 route exists and is reachable from the local device.

For example, you can configure HSRP to track the line protocol of the interface that connects one of the redundant routers to the rest of the network. If that link protocol goes down, you can modify the priority of the affected HSRP router and cause a switchover to a backup router that has better network connectivity.

For more information related to object tracking, see the following chapters:

- #unique_778

- #unique_779

- Configuring Layer 3 Virtualization

Object Track List

An object track list allows you to track the combined states of multiple objects. Object track lists support the following capabilities:

- Boolean "and" function—Each object defined within the track list must be in an up state so that the track list object can become up.

- Boolean "or" function—At least one object defined within the track list must be in an up state so that the tracked object can become up.

- Threshold percentage—The percentage of up objects in the tracked list must be greater than the configured up threshold for the tracked list to be in the up state. If the percentage of down objects in the tracked list is above the configured track list down threshold, the tracked list is marked as down.

- Threshold weight—Assign a weight value to each object in the tracked list, and a weight threshold for the track list. If the combined weights of all up objects exceeds the track list weight up threshold, the track list is in an up state. If the combined weights of all the down objects exceeds the track list weight down threshold, the track list is in the down state.

Other entities, such as virtual Port Channels (vPCs) can use an object track list to modify the state of a vPC based on the state of the multiple peer links that create the vPC.

See the *Cisco Nexus 7000 Series NX-OS Interfaces Configuration Guide*, for more information on vPCs.

High Availability

Object tracking supports high availability through stateful restarts. A stateful restart occurs when the object tracking process crashes. Object tracking also supports a stateful switchover on a dual supervisor system. Cisco NX-OS applies the runtime configuration after the switchover.

You can also use object tracking to modify the behavior of a client to improve overall network availability.

Virtualization Support

Object tracking supports Virtual Routing and Forwarding (VRF) instances. VRFs exist within virtual device contexts (VDCs). By default, Cisco NX-OS places you in the default VDC and default VRF unless you specifically configure another VDC and VRF. By default, Cisco NX-OS tracks the route reachability state of objects in the default VRF. If you want to track objects in another VRF, you must configure the object to be a member of that VRF.

For more information, see the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*, and Configuring Layer 3 Virtualization.

Prerequisites for Object Tracking

Note   For a full list of feature-specific prerequisites, see the platform-specific documentation.

Guidelines and Limitations for Object Tracking

Object Tracking has the following configuration guidelines and limitations:

- Supports up to 500 tracked objects per VDC.

- Supports Ethernet, subinterfaces, tunnels, port channels, loopback interfaces, and VLAN interfaces.

- Supports one tracked object per HSRP group or GLBP group.

- If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Default Settings for Object Tracking Parameters

Default Object Tracking Parameters

Parameters	Default
Tracked Object VRF	Member of default VRF

Configuring Object Tracking

Configuring Object Tracking for an Interface

You can configure Cisco NX-OS to track the line protocol or IPv4 or IPv6 routing state of an interface.

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **track** *object-id* **interface** *interface-type number* {{**ip**	**ipv6**} **routing**	**line-protocol**}	Creates a tracked object for an interface and enters tracking configuration mode. The object-id range is from 1 to 500.
Step 3	(Optional) switch(config-track)# **show track** [*object-id*]	Displays object tracking information.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to configure object tracking for the line protocol state on Ethernet 1/2:

```
switch # configure terminal
switch(config)# track 1 interface ethernet 1/2 line-protocol
switch(config)# copy running-config startup-config
```

The following example shows how to configure object tracking for the IPv4 routing state on Ethernet 1/2:

```
switch # configure terminal
switch(config)# track 2 interface ethernet 1/2 ip routing
switch(config)# copy running-config startup-config
```

The following example shows how to configure object tracking for the IPv6 routing state on Ethernet 1/2:

```
switch # configure terminal
switch(config)# track 3 interface ethernet 1/2 ipv6 routing
switch(config)# copy running-config startup-config
```

Deleting a Tracking Object

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **no track** *1*	Deletes a tracked object for an interface. The object-id range is from 1 to 500.
Step 3	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to delete an object tracking:

```
switch # configure terminal
switch(config)# no track 1
switch(config)# copy running-config startup-config
```

Configuring Object Tracking for Route Reachability

You can configure Cisco NX-OS to track the existence and reachability of an IP route or IPv6 route.

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **track** *object-id* {**ip**	**ipv6**} **route** *prefix/length* **reachability**	Creates a tracked object for a route and enters tracking configuration mode. The object-id range is from 1 to 500. The prefix format for IP is A.B.C.D/length, where the length range is from 1 to 32. The prefix format for IPv6 is A:B::C:D/length, where the length range is from 1 to 128. The prefix format for IPv6 is A:B::C:D/length, where the length range is from 1 to 128.

	Command or Action	**Purpose**
Step 3	(Optional) switch(config-track)# **show track** [*object-id*]	Displays object tracking information.
Step 4	switch(config-track)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying and running configuration to the startup configuration.

Example

The following example shows how to configure object tracking for an IPv4 route in the default VRF:

```
switch # configure terminal
switch(config)# track 4 ip route 192.0.2.0/8 reachability
switch(config-track)# copy running-config startup-config
```

The following example shows how to configure object tracking for an IPv6 route in the default VRF:

```
switch # configure terminal
switch(config)# track 5 ipv6 route 10::10/128 reachability
switch(config-track)# copy running-config startup-config
```

Configuring an Object Track List with a Boolean Expression

You can configure an object track list that contains multiple tracked objects. A tracked list contains one or more objects. The Boolean expression enables two types of calculation by using either "and" or "or" operators. For example, when tracking two interfaces using the "and" operator, up means that both interfaces are up, and down means that either interface is down.

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **track** *track-number* **list boolean** {**and** | **or**}	Configures a tracked list object and enters tracking configuration mode. Specifies that the state of the tracked list is based on a Boolean calculation. The keywords are as follows: • and—Specifies that the list is up if all objects are up or down if one or more objects are down. For example, when tracking two interfaces, up means that both interfaces are up, and down means that either interface is down. • or—Specifies that the list is up if at least one object is up. For example, when

	Command or Action	Purpose
		tracking two interfaces, up means that either interface is up, and down means that both interfaces are down.
		The track-number range is from 1 to 500.
Step 3	switch(config-track)# **object** *object-number* [**not**]	Adds a tracked object to the track list. The object-id range is from 1 to 500. The not keyword optionally negates the tracked object state.
		Note The example means that when object 10 is up, the tracked list detects object 10 as down.
Step 4	(Optional) switch(config-track)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 5	(Optional) switch(config-track)# **show track**	Displays object tracking information.

Example

The following example shows how to configure a track list with multiple objects as a Boolean "and":

```
switch # configure terminal
switch(config)# track 1 list boolean and
switch(config-track)# object 10
switch(config-track)# object 20 not
switch(config)# copy running-config startup-config
```

Configuring an Object Track List with a Percentage Threshold

You can configure an object track list that contains a percentage threshold. A tracked list contains one or more objects. The percentage of up objects must exceed the configured track list up percent threshold before the track list is in an up state. For example, if the tracked list has three objects, and you configure an up threshold of 60 percent, two of the objects must be in the up state (66 percent of all objects) for the track list to be in the up state.

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# **track** *track-number* **list threshold percentage**	Configures a tracked list object and enters tracking configuration mode. Specifies that the state of the tracked list is based on a configured threshold percent. The track-number range is from 1 to 500.
Step 3	switch(config-track)# **threshold percentage up** *up-value* **down** *down-value*	Configures the threshold percent for the tracked list. The range from 0 to 100 percent.
Step 4	switch(config-track)# **object** *object-number*	Adds a tracked object to the track list. The object-id range is from 1 to 500. **Note** The example means that when object 10 is up, the tracked list detects object 10 as down.
Step 5	(Optional) switch(config-track)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 6	(Optional) switch(config-track)# **show track**	Displays object tracking information.

Example

The following example shows how to configure a track list with multiple objects as a Boolean "and":

```
switch # configure terminal
switch(config)# track 1 list threshold percentage
switch(config-track)# threshold percentage up 70 down 30
switch(config-track)# object 10
switch(config-track)# object 20
switch(config-track)# object 30
switch(config-track)# copy running-config startup-config
```

Configuring an Object Track List with a Weight Threshold

You can configure an object track list that contains a weight threshold. A tracked list contains one or more objects. The combined weight of up objects must exceed the configured track list up weight threshold before the track list is in an up state. For example, if the tracked list has three objects with the default weight of 10 each, and you configure an up threshold of 15, two of the objects must be in the up state (combined weight of 20) for the track list to be in the up state.

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **track** *track-number* **list threshold weight**	Configures a tracked list object and enters tracking configuration mode. Specifies that the state of the tracked list is based on a configured threshold weight. The *track-number* range is from 1 to 500.
Step 3	switch(config-track)# **threshold weight up** *up-value* **down** *down-value*	Configures the threshold weight for the tracked list. The range is from 1 to 255.
Step 4	switch(config-track)# **object** *object-id* **weight** *value*	Adds a tracked object to the track list. The *object-id* range is from 1 to 500. The *value* range is from 1 to 255. The default weight value is 10
Step 5	switch(config-track)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 6	(Optional) switch(config-track)# **show track**	Displays object tracking information.

Example

The following example shows how to configure a track list with a up weight threshold of 30 and a down threshold of 10:

> **Note** In this example, the track list is up if object 10 and object 20 are up, and the track list goes to the down state if all three objects are down.

```
switch # configure terminal
switch(config)# track 1 list threshold weight
switch(config-track)# threshold weight up 30 down 10
switch(config-track)# object 10 weight 15
switch(config-track)# object 20 weight 15
switch(config-track)# object 30
switch(config-track)# copy running-config startup-config
```

Configuring an Object Tracking Delay

You can configure a delay for a tracked object or an object track list that delays when the object or list triggers a stage change. The tracked object or track list starts the delay timer when a state change occurs but does not recognize a state change until the delay timer expires. At that point, Cisco NX-OS checks the object state again and records a state change only if the object or list currently has a changed state. Object tracking ignores any intermediate state changes before the delay timer expires.

For example, for an interface line-protocol tracked object that is in the up state with a 20 second down delay, the delay timer starts when the line protocol goes down. The object is not in the down state unless the line protocol is down 20 seconds later.

You can configure independent up delay and down delay for a tracked object or track list. When you delete the delay, object tracking deletes both the up and down delay.

You can change the delay at any point. If the object or list is already counting down the delay timer from a triggered event, the new delay is computed as the following:

- If the new configuration value is less than the old configuration value, the timer starts with the new value.

- If the new configuration value is more than the old configuration value, the timer is calculated as the new configuration value minus the current timer countdown minus the old configuration value.

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **track** *object-id* {*parameters*}	Creates a tracked object for a route and enters tracking configuration mode. The object-id range is from 1 to 500. The prefix format for IP is A.B.C.D/length, where the length range is from 1 to 32. The prefix format for IPv6 is A:B::C:D/length, where the length range is from 1 to 128.
Step 3	switch(config-track)# **track** *track-number* **list** {*parameters*}	Configures a tracked list object and enters tracking configuration mode. Specifies that the state of the tracked list is based on a configured threshold weight. The *track-number* range is from 1 to 500.
Step 4	switch(config-track)# **delay** {**up** *up-time* [**down** *down-time*] | **down** *down-time* [**up** *up-time*]}	Configures the object delay timers. The range is from 0 to 180 seconds. The *track-number* range is from 1 to 500.
Step 5	(Optional) switch(config-track)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 6	(Optional) switch(config-track)# **show track**	Displays object tracking information.

Example

The following example shows how to configure object tracking for a route and use delay timers:

```
switch # configure terminal
switch(config)# track 2 ip route 209.165.201.0/8 reachability
switch(config-track)# delay up 20 down 30
switch(config-track)# copy running-config startup-config
```

The following example shows how to configure a track list with an up weight threshold of 30 and a down threshold of 10 with delay timers:

```
switch# configure terminal
switch(config)# track 1 list threshold weight
switch(config-track)# threshold weight up 30 down 10
switch(config-track)# object 10 weight 15
switch(config-track)# object 20 weight 15
switch(config-track)# object 30
switch(config-track)# delay up 20 down 30
switch(config-track)# copy running-config startup-config
```

The following example shows the delay timer in the **show track** command output before and after an interface is shut down:

```
switch(config-track)# show track
Track 1
  Interface loopback1 Line Protocol
  Line Protocol is UP
  1 changes, last change 00:00:13
  Delay down 10 secs

switch(config-track)# interface loopback 1
switch(config-if)# shutdown
switch(config-if)# show track
Track 1
  Interface loopback1 Line Protocol
  Line Protocol is delayed DOWN (8 secs remaining)<------- delay timer counting down
  1 changes, last change 00:00:22
  Delay down 10 secs
```

Configuring Object Tracking for a Nondefault VRF

You can configure Cisco NX-OS to track an object in a specific VRF.

Before you begin

- Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

- Ensure that nondefault VRFs are created first.

Procedure

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **track** *object-id* {**ip** | **ipv6**} **route** *prefix/length* **reachability**	Creates a tracked object for a route and enters tracking configuration mode. The *object-id* range is from 1 to 500. The prefix format for IP is A.B.C.D/length, where the length range is from 1 to 32. The prefix format for IPv6 is

	Command or Action	Purpose
		A:B::C:D/length, where the length range is from 1 to 128.
Step 3	switch(config-track)# **vrf member** *vrf-name*	Configures the VRF to use for tracking the configured object.
Step 4	switch(config-track)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 5	(Optional) switch(config-track)# **show track**	Displays object tracking information.

Example

The following example shows how to configure object tracking for a route and use VRF Red to look up reachability information for this object:

```
switch # configure terminal
switch(config)# track 2 ip route 209.165.201.0/8 reachability
switch(config-track)# vrf member Red
switch(config-track)# copy running-config startup-config
```

The following example shows how to configure object tracking for an IPv6 route and use VRF Red to look up reachability information for this object:

```
switch# configure terminal
switch(config)# track 3 ipv6 route 1::2/64 reachability
switch(config-track)# vrf member Red
switch(config-track)# copy running-config startup-config
```

The following example how to modify tracked object 2 to use VRF Blue instead of VRF Red to look up reachability information for this object:

```
switch(config-track)# show track
switch(config)# track 2
switch(config-track)# vrf member Blue
switch(config-track)# copy running-config startup-config
```

Verifying the Object Tracking Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show track [*object-id*] [**brief**]	Displays the object tracking information for one or more objects.
show track [*object-id*] **interface** [**brief**]	Displays the interface-based object tracking information.

Command	Purpose
show track [*object-id*] {**ip** | **ipv6**} **route** [**brief**]	Displays the IPv4 or IPv6 route-based object tracking information.
show trun track	Displays the IP route IPv6 object tracking configuration information.

Configuration Example for Object Tracking

This example shows how to configure object tracking for route reachability and use VRF Red to look up reachability information for this route:

```
switch# configure terminal
switch(config)# track 2 ip route 209.165.201.0/8 reachability
switch(config-track)# vrf member Red
```

Related Documents for Object Tracking

Related Topic	Document Title
Object Tracking CLI commands	*Cisco Nexus 7000 Series NX-OS Unicast Routing Command Reference*
Configuring the Embedded Event Manager	*Cisco Nexus 7000 Series NX-OS System Management Configuration Guide*

Standards for Object Tracking

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

Feature History for Object Tracking

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Feature Name	Releases	Feature Information
IPv6	5.0(2)	Added support for IPv6.
Tracking delay	4.2(4)	Added support for delaying a tracked object update.
Object track list	4.2(1)	Added support for object track lists and Boolean expressions.
Object tracking	4.0(1)	This feature was introduced.