Configuring SRv6 Traffic Engineering

This chapter contains information on how to configure SRv6 traffic engineering.

About SRv6 Traffic Engineering

SRv6 traffic engineering (SRv6 TE) uses the concept of source routing, where the source calculates the path and encodes it in the packet header as a list of segments. This list of segments is added to an IPv6 routing header called the SRv6 Segment Routing Header (SRH) in the incoming packet.

With SRv6 TE, the network does not need to maintain per-application and per-flow state on each node. Instead only the head-end nodes on the edge of the network where the traffic enters the policy need to maintain state. The remaining nodes simply obey the forwarding instructions that are provided in the packet.

SRv6 traffic engineering can utilize network bandwidth more effectively than traditional MPLS RSVP-TE by using ECMP within each segment. In addition, by using a single intelligent source that it relieves remaining routers from the task of calculating the required path through the network.

SRv6 Traffic Engineering Policies

SRv6 traffic engineering uses a “policy” to steer traffic through the network. A SRv6 traffic engineering policy is a container that includes sets of segments.

The headend imposes SID list on traffic flow. Each transit node in the SID stack uses the top SID to choose the next-hop, pops the SID, and forwards the packet to the next node. The packet is forwarded with the remainder of the SID stack, until it reaches the ultimate destination.

A SRv6 traffic engineering policy is uniquely identified by a tuple (color, endpoint). Color is represented as a 32-bit number while the IPv6 address is an endpoint. Every SRv6 traffic engineering policy has a color value. Every policy between the same node pairs requires unique color value. Multiple SRv6 traffic engineering policies can be created between the same two endpoints by choosing different colors for these policies.

In Cisco NX-OS Release 9.3(5), Cisco Nexus 9000 Series switches support only explicit SRv6 policy.

Explicit SRv6 Traffic Engineering Policy

An explicit policy is a list of IPv6 addresses representing an ordered list of segment IDs. The policy path is statically configured because the segment list is defined by the operator.

To create an explicit policy, you must first define segment list (s), the policy name, endpoint, and color and reference it to a segment list from the policy. Segment lists are defined separately since these can be reused between different policies.

Currently, the list of segments in an explicit policy must contain only the SRv6 END SIDs of the nodes in the path (excluding the headend). Each policy supports a maximum of three preferences; three segment lists where only one is active at any given point. This allows you to have one active segment list and two backup segment lists.

Destination Prefix Based Traffic Steering

Global VRF

You can configure a destination prefix and a prefix length in the global VRF and steer it through a SRv6 traffic engineering policy. This destination prefixes can be either IPv4 or IPv6 addresses. A policy can be referenced for traffic engineering based on the policy name or the color and the endpoint. If the destination prefix is an IPv6 prefix which is reachable via the IGP, BGP, or static without any SRv6 encapsulation, the traffic steering occurs with a T.insert behavior with the SIDs in the SRH. In this case, the traffic engineered route takes precedence over the original best route in the forwarding.

If the destination prefix is an IPv4 or IPv6 prefix which is reachable via an SRv6 encapsulation, the traffic steering occurs with the T.encap behavior. The remote encapsulation is inherited from the remote global VRF over SRv6. The traffic engineered path is derived from a SRv6 traffic engineering policy. In this case, the final traffic engineered route takes precedence over the original T.encap route in forwarding.

You can configure a complete encapsulation without a SRv6 traffic engineering policy. In this case, encapsulation that is configured by you takes precedence over remote learned remote routes.

VPN VRF

You can configure a destination prefix and a prefix length in a VPN VRF and steer it through a SRv6 traffic engineering policy. This destination prefix can be IPv4 or IPv6 addresses. A policy can be referenced for traffic engineering based on the policy name or the color and the endpoint.

If the destination prefix is an IPv4 and IPv6 prefix and is learned from BGP, the remote encapsulation is inherited from the remote VPN route. The traffic engineering path is derived from a SRv6 traffic engineering policy. The final traffic engineering SIDs with T.Encap take precedence over the original best route in the forwarding.

You can configure a complete encapsulation without a SRv6 traffic engineering policy. In this case, encapsulation that is configured by you takes precedence over remote learned remote routes.

Guidelines and Limitations for SRv6 Traffic Engineering

SRv6 traffic engineering has the following guidelines and limitations:

- Beginning with Cisco NX-OS Release 9.3(3), SRv6 traffic engineering is supported on Cisco Nexus 9300-GX and 9300-GX2 platform switches.
- In Cisco NX-OS Release 9.3(5), only one tunnel profile is supported.
- The maximum number of SRv6 SIDs in the SR-TE path with T.Encaps is 4.
- The maximum number of SRv6 SIDs in the SR-TE path with T.Insert is 8.
- ECMP is not supported at the policy level. There is only one path per preference in the SR-TE. Maximum of three preferences are supported.
- The MPLS segment routing and SRv6 features cannot be enabled concurrently.
- IPv6 redirects must not be configured on core interfaces. Use the **no ipv6 redirects** command to disable IPv6 redirects.

Creating the Explicit SID List

You can create segment-list and explicit SRv6 traffic engineering policy.

Before you begin

You must ensure that the SRv6 feature is enabled.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**segment-routing**
	Example:
	`switch(config)#segment-routing`
	`switch(config-sr)#`
		Enters the segment routing configuration mode.

Step 3	**srv6**
	Example:
	`switch(config)#srv6`
	`switch(config-sr-srv6)#`
		Enables segment routing over SRv6.

Step 4	**traffic-engineering** **Example:**	Enters the traffic engineering mode.

	Command or Action	Purpose
	`switch(config-sr-srv6)# traffic-engineering` `switch(config-sr-srv6-te)#`	

Step 5	**segment-list name** *sidlist-name*
	Example:
	`switch(config-sr-srv6-te)# segment-list name black`
	`index 1 segment-routing srv6 A1:0:0:2:1::`
	`index 5 segment-routing srv6 A1:0:0:3:1::`
	`segment-list name blue`
	`index 1 segment-routing srv6 A1:0:0:4:1::`
	`index 5 segment-routing srv6 A1:0:0:5:1::`
		Creates the explicit SID list.

Step 6	**policy** *policy name* **Example:** `switch(config-sr-te-color)# policy 1`	Configures the policy.
Step 7	**color** *numberIPv6-end-point* **Example:** `switch(config-sr-te-pol)# color 201 endpoint A1:0:0:07::1`	Configures the color and the endpoint of the policy.

Step 8	**candidate-paths**
	Example:
	`switch(config-sr-te-color)# candidate-paths`
	`switch(cfg-cndpath)#`
		Specifies the candidate paths for the policy.

Step 9	**preference** *preference-number*
	Example:
	`switch(cfg-cndpath)# preference 100`
	`switch(cfg-pref)#`
		Specifies the preference of the candidate path.

Step 10	**explicit segment-list** *sidlist-name*
	Example:
	`switch(cfg-dyn)# explicit segment-list blue`
	`switch(cfg-dyn)#`
		Specifies that the explicit list.

Step 11	**exit**
	Example:
	`switch(cfg-dyn)# exit`
	`switch(config)#`
		Exits the configuration mode.

	Command or Action	Purpose

Step 12	**srv6**
	Example:
	`switch(config)# srv6`
	`switch(config-srv6)#`
		Enters the SRv6 configuration mode.

Step 13	**locators**	Enters the locators configuration.
Step 14	**locator** *name*	Configures the locator name, which is the global locator name that was globally configured for SRv6.

Associating Prefixes to an Explicit SRv6 Traffic Engineering Policy

You can contain the source IPv6 address using the SRv6 encapsulation configuration.

Before you begin

Ensure that **feature srv6** is enabled.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**feature ofm** **Example:** `switch (config)# feature ofm`	Enables ofm.
Step 3	**tunnel profile** *main* **Example:** `switch(config-sr-srv6)# tunnel profile main`	Creates the tunnel profile for SRv6 encapsulation.

Step 4	**encapsulation srv6**
	Example:
	`switch(config-tnl-profile)# encapsulation srv6`
	`switch(config-tnl-profile)#`
		Creates a tunnel profile for SRv6.

Step 5	**route** *prefix* / *len* [**vrf** *vpm-vrf*] **via policy color** *color* **endpoint** *endpoint address*	Associates the prefix to the policy.

	Command or Action	Purpose
	Example: `switch(config-sr-srv6-encap)# route 10.1.1.2/32 vrf vrf1 via policy BLUE_PATH`	

Configuration Example for SRv6 Traffic Engineering

This example shows the SRv6 traffic engineering configuration:

```
segment-routing
    traffic-engineering
        srv6
            locator main
    segment-list name black
        index 1 A1:0:0:2:1::
        index 5 A1:0:0:3:1::
    segment-list name blue
        index 1 A1:0:0:4:1::
        index 5 A1:0:0:5:1::
    policy policy1
        color 201 endpoint A1:0:0:07::1
        candidate-paths
            preference 70
                explicit segment-list black
            preference 100
                explicit segment-list blue
```

Examples of configuring prefixes for SRv6 traffic engineering. The VRF name variable (vrf_name) can be global or default, or the L3VPN VRF.

```
tunnel-profile main
    encapsulation srv6

  route vrf <vrf_name> 3.0.1.0/24  via policy name POLICY1
  route vrf <vrf_name> 3::1:0/124  via policy name POLICY1

  route vrf <vrf_name> 3.0.2.0/24  via policy color 1 endpoint fd00::a02:2
  route vrf <vrf_name> 3::2:0/124  via policy color 1 endpoint fd00::a02:2

  route vrf <vrf_name> 3.0.3.0/24 remote-locator fd01:0:0:2:: function 65533
  route vrf <vrf_name> 3::3:0/124 remote-locator fd01:0:0:2::  function 65533

  route vrf <vrf_name> 3.0.4.0/24 remote-locator fd01:0:0:2:: function 65533 via policy
color 1 endpoint fd00::a02:2
  route vrf <vrf_name> 3::4:0/124 remote-locator fd01:0:0:2:: function 65533 via policy
color 1 endpoint fd00::a02:2

  route vrf <vrf_name> 3.0.5.0/24 remote-locator fd01:0:0:3:: function 65533 via policy
name POLICY1
  route vrf <vrf_name> 3::5:0/124 remote-locator fd01:0:0:3:: function 65533 via policy
name POLICY1
```

Verifying SRv6 Traffic Engineering Configuration

To display the SRv6 traffic engineering configuration, perform one of the following tasks:

Command	Purpose
show running srte	Displays the SRv6 traffic engineering configuration.
show running ofm	Displays the static route configuration.