

Segment Routing over IPv6

Segment Routing (SR) can be applied on both MPLS and IPv6 dataplanes. From Cisco IOS XE 17.12.1a, Segment Routing over IPv6 (SRv6) extends Segment Routing support over the IPv6 dataplane.

- Segment Routing over IPv6, on page 2
- Configuring SRv6, on page 6
- SRv6 under IS-IS, on page 10
- SRv6 BGP-Based Services, on page 12
- EVPN VPWS over SRv6, on page 25
- BGP SRv6 L3VPN On-Demand Next-Hop, on page 36
- SRv6 Traffic Engineering Policies, on page 46
- Performance Measurement for SRv6, on page 53
- SRv6 OAM, on page 59

Segment Routing over IPv6

Feature Information

Table 1: Feature Information Table for SRv6

Feature Name	Release	Description
Segment Routing over IPv6 Dataplane	Cisco IOS XE Release 17.12.1a	Segment Routing (SR) can currently be applied on Multiprotocol Label Switching (MPLS) dataplane. From Cisco IOS XE 17.12.1a, SR is supported over the IPv6 dataplane for the following protocols: • Interior Gateway Protocol (IS-IS only) • Border Gateway Protocol (BGP) In addition, the following functionalities are available for Segment Routing over IPv6 dataplane: • Segment Routing Traffic Engineering Policies • Static Routes • Performance Management • Operations, Administration and Maintenance (OAM)
BGP SRv6L3VPN On-Demand Next-Hop	CiscoIOS XE Release 17.13.1a	This feature was introduced.
Enhancements to SRv6	CiscoIOS XE Release 17.16.1a	The following features were introduced: • eBGP Inter-AS • PCE-Delegated Path Computation • Enhancements to OAM Traffic Engineering

Note For information about supported platforms for each feature and release, see Supported Platforms, on page 6.

Restrictions for SRv6

- Cisco IOS XE supports uSIDs with 32-bit uSID block and 16-bit uSID IDs (3216). This format must be used for uSID locators in a SRv6 uSID domain.
- Cisco IOS XE supports up to 10 uSID locators.
- Cisco IOS XE supports the following SRv6 uSID behaviors and variants:
  - uN with PSP/USD
  - uA with PSP/USD
  - uDT4
  - uDT6
  - uDT46
- Cisco IOS XE supports **H.Encaps.Red** SRv6 policy headend behavior.

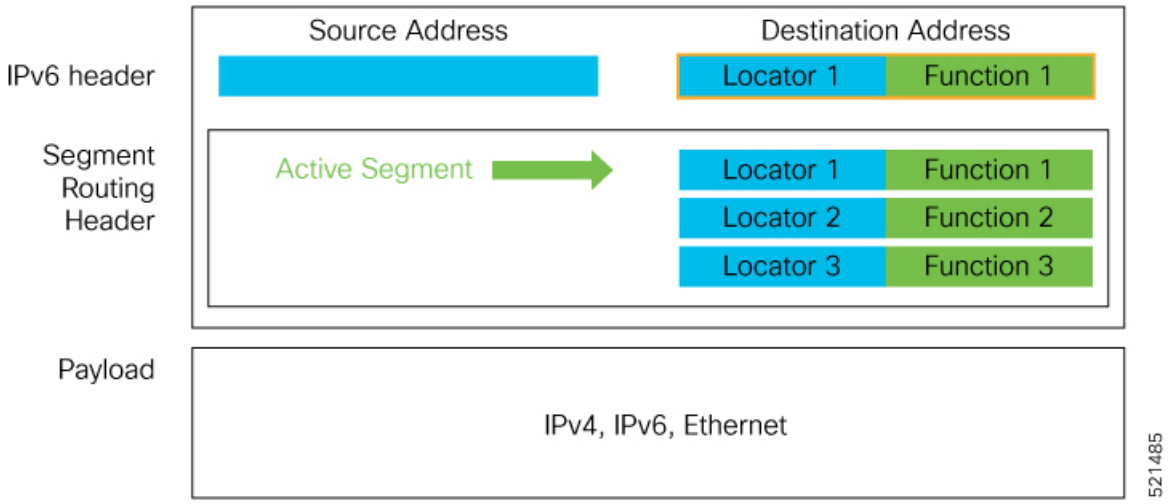
Information About SRv6

In an SR-MPLS enabled network, an MPLS label represents an instruction. The source nodes program the path to a destination in the packet header as a stack of labels.

SRv6 introduces the Network Programming framework that enables a network operator or an application to specify a packet processing program by encoding a sequence of instructions in the IPv6 packet header. Each instruction is implemented on one or several nodes in the network and identified by an SRv6 Segment Identifier (SID) in the packet. The SRv6 Network Programming framework is defined in IETF RFC 8986 SRv6 Network Programming.

In SRv6, an IPv6 address represents an instruction. SRv6 uses a new type of IPv6 Routing Extension Header, called the Segment Routing Header (SRH), in order to encode an ordered list of instructions. The active segment is indicated by the destination address of the packet, and the next segment is indicated by a pointer in the SRH.

Figure 1: Network Program in the Packet Header



The SRv6 SRH is documented in IETF RFC 8754 IPv6 Segment Routing Header (SRH).

SRv6 Node Roles

The SRv6 Node Roles are documented in the IETF RFC 8754 IPv6 Segment Routing Header (SRH).

SRv6 Head-End Behaviors

The SRv6 Head-end with Encapsulation behaviors are documented in the IETF RFC 8986 SRv6 Network Programming.

SRv6 Endpoint Behaviors

The SRv6 endpoint behaviors are documented in the IETF RFC 8986 SRv6 Network Programming.

SRv6 Endpoint Behavior Variants

The SRv6 endpoint behavior variants are documented in the IETF RFC 8986 SRv6 Network Programming.

SRv6 Micro-Segment (uSID)

Several SRv6 uSIDs may be encoded within a single 128-bit SID, called a uSID carrier.

SRv6 uSID is documented in the IETF drafts NetworkProgramming extension: SRv6 uSID instruction and Compressed SRv6 Segment List Encoding in SRH.

Throughout this chapter, SRv6 micro-segment is referred to as **uSID**.

SRv6 uSID Terminology

The SRv6 uSID terminology is documented in the Network Programming extension: SRv6 uSID instruction.

SRv6 uSID Allocation Within a uSID Block

SRv6 uSID allocations are documented in the Network Programming extension: SRv6 uSID instruction.

SRv6 Endpoint Behaviors Associated with uSID

SRv6 uSID endpoint behaviors are documented in the Network Programming extension: SRv6 uSID instruction.

SRv6 Implementation

A new command **segment-routing srv6** is introduced in Cisco IOS XE 17.12.1a to enable SRv6 configuration.

```
segment-routing srv6
    encapsulation
        source-address {ipv6-addr}
        hop-limit [propagate | <value>]
        traffic-class [propagate | <value>]
    locators
        locator <name>
          format usid-f3216
          prefix <locator-ipv6-prefix/prefix-len>
    sid holdtime <value>
    explicit-sids
        sid <SRv6-SID> behavior {end-dt46 | end-dt4 | end-dt6}
            forwarding
                path <1-1>
                    decap-and-lookup [vrf-name <vrf>]
```

The parameters for this command are described below.

SRv6 Locator Name, Prefix, and uSID-Related Parameters

This section describes the configurable keywords for the **segment-routing SRv6** command.

locator *name*	Configures the SRv6 locator.
locator *name* **prefix** *locator*	Configures the locator prefix value.
locator *name* **format** *usid-f3216*	Specifies the locator as a micro-segment (uSID).

SRv6 Encapsulation Parameters

This section describes the configurable SRv6 encapsulation parameters. These optional parameters include:

encapsulation source-address ipv6-addr	Source Address of outer encapsulating IPv6 header. The default source address for encapsulation is the lowest global unicast IPv6 address of lowest loopback interface. If loopback addresses and static encapsulation source address are not configured, the source address remains unassigned (0::0).
encapsulation hop-limit {*count* | <*propagate*>}	The hop limit of outer-encapsulating IPv6 header. The range for *count* is from 1 to 255; the default value for hop-limit is 64. Use **propagate** to set the hop-limit value by propagation (from incoming packet/frame).

encapsulation traffic-class {*value* | <*propagate*>}	The traffic-class field settings on the IPv6 header. Specify the *value* (as 2 hexadecimal nibbles) for traffic class; valid values are from 0x0 to 0xff. The default value is 0. Use **propagate** to set the traffic-class value by propagation (from incoming packet/frame).

SRv6 SID Parameters

This section describes the configurable SRv6 SID parameters.

sid holdtime *minutes*	The hold time for a stale or freed SID. The range of *minutes* is from 0 (disabled) to 60 minutes.
sid <*SRv6-SID*> **behavior** {*end-dt46* | *end-dt4* | *end-dt6*}	Configures a static SID, given the SID address and the behavior context.

Supported Platforms

From Cisco IOS XE 17.12.1a release, several SRv6 features are supported on the following platforms:

- Cisco ASR1000 RP3 + ESP100-X/ESP200-X, ASR1001-HX, ASR1002-HX
- Cisco Catalyst 8000V Edge Software
- Cisco Catalyst 8200 Series Edge platforms
- Cisco Catalyst 8300 Series Edge platforms
- Cisco Catalyst 8500 and 8500L Series Edge platforms

For information about the supported features, see Feature Information, on page 2.

From Cisco IOS XE 17.13.1a release, BGP SRv6 L3VPN ODN is supported on the following platforms:

- Cisco ASR1000 RP3 + ESP100-X/ESP200-X, ASR1001-HX, ASR1002-HX
- Cisco Catalyst 8000V Edge Software
- Cisco Catalyst 8200 Series Edge platforms
- Cisco Catalyst 8300 Series Edge platforms
- Cisco Catalyst 8500 and 8500L Series Edge platforms

Configuring SRv6

Configuring SRv6

Enabling SRv6 involves the following high-level configuration steps:

- Configuring Global SRv6 with locator(s)
- Configuring Optional SRv6 Parameters

Configuring Global SRv6 with a Locator

The following example shows how to globally enable SRv6 and configure a locator:

```
Router(config)# segment-routing srv6
Router(config-srv6)# locators
Router(config-srv6-locator)# locator myLoc1
Router(config-srv6-locator)# format usid-f3216
Router(config-srv6-locator)# prefix 2001:0:8::/48
```

Configuring Optional SRv6 Parameters

The following example shows how to configure optional SRv6 parameters:

```
Router(config)# segment-routing srv6
Router(config-srv6)# encapsulation
Router(config-srv6-encap)# source-address 1::1
Router(config-srv6-encap)# hop-limit 60
Router(config-srv6-encap)# traffic-class propagate
Router(config-srv6-encap)# exit
Router(config-srv6)# sid holdtime 10
```

Configuring SRv6 Static Explicit END.X SID

From Cisco IOS XE 17.15.1a, you can configure static explicit END.X:

```
segment-routing srv6
      explicit-sids
         sid <sid> behavior end-x-usid-psp-usd
            identification nexthop [interface] <IPv6 address> <protected>
```

Note IPv6 address is not required for point-to-point interface. Use link-local address to specify IPv6 address.

Configuring SRv6 Path MTU

From Cisco IOS XE 17.15.1a, a new global static configuration of MTU for all SRv6 policy and VPN-SID is available:

```
segment-routing srv6
  encapsulation
     mtu <1280-9960>
```

Verifying SRv6 Configuration

Use the following examples to verify SRv6 configuration.

Example 1: This example shows how to verify the locator configuration and its operational status:

```
router# show segment-routing srv6 locator
Name            Algo     Prefix                    Format              Status
----            ----     ------                    ------              ------
loc1            0        FC01:101:2::/48           usid-f3216          Up
```

Example 2: The following examples show how to view the platform capabilities and parameters:

```
router# show segment-routing srv6 capabilities-parameters
Platform Capabilities:
 SRv6:Yes
 PFP:Yes
 TILFA:No
 Endpoint behaviors:
  uN (PSP/USD)
  uA (PSP/USD)
  uDT6
  uDT4
  uDT46
  Transit.ENCAP.RED
 Encap Parameters:
  Max-SL :16
  Encap :Collapsed
  Hop-limit propagate :Yes
  Traffic-class propagate :Yes
Parameters in-use:
 Encap Parameters:
  Source Address: 2001::1:1:1:2, Loopback1 (Default)
  Hop-Limit: 64 (Default)
  Traffic-class: 0 (Default)
```

```
router# show srv6 capabilities-parameters
Platform Capabilities:
 SRv6:Yes
 PFP:Yes
 TILFA:No
 Endpoint behaviors:
  uN (PSP/USD)
  uA (PSP/USD)
  uDT6
  uDT4
  uDT46
  Transit.ENCAP.RED
 Encap Parameters:
  Max-SL :16
  Encap :Collapsed
  Hop-limit propagate :Yes
  Traffic-class propagate :Yes
Parameters in-use:
  Encap Parameters:
    Source Address: A001::1, Loopback0 (Default)
    Hop-Limit: 64 (Default)
    Traffic-class: 0 (Default)
```

Example 3: The following examples show how to view the SID overview and details:

```
router# show segment-routing srv6 sid
SID                  Locator      Behavior          Context               Owner
---                  -------      --------          -------               -----
FC01:101:2::          loc1        uN (PSP/USD)                                    SID-MGR

FC01:101:2:E000::     loc1         uDT4             ce1                             router
 bgp
FC01:101:2:E001::     loc1         uDT6             ce1                             router
 bgp
FC01:101:2:E002::     loc1        uA (PSP/USD)      Ethernet2/0 2001::99:2:3:3      router
 isis sr
FC01:101:2:E003::     loc1        uA (PSP/USD)      Ethernet2/1 2001::100:2:3:3     router
 isis sr
FC01:101:2:E004::     loc1        uA (PSP/USD)      Ethernet3/0 2001::99:2:4:4      router
 isis sr
FC01:101:2:E005::     loc1        uA (PSP/USD)      Ethernet3/1 2001::100:2:4:4     router
```

```
 isis sr
FC01:101:2:E006::     loc1         uA (PSP/USD)      Ethernet4/0 2001::99:2:5:5       router
 isis sr
FC01:101:2:E007::     loc1         uA (PSP/USD)      Ethernet4/1 2001::100:2:5:5      router
 isis sr
```

```
router# show segment-routing srv6 sid FC01:101:2:: detail
SID: FC01:101:2::   Type: DYNAMIC
 Behavior: uN (PSP/USD) (48)
 Context:
    interface: (not-set)
    vrf: (not-set), v4-topo-id: 0xFFFF, v6-topo-id: 0xFFFF
    next-hop: (not-set)
    policy: (not-set)
    distinguisher: (not-set)
 Stats:
    Packets: 0  Bytes: 0
 User list:
    User:Refcount        Locator:Refcount
    -------------        ----------------
    SID-MGR(2):1         loc1:1
 Event history:
    Timestamp            Client           Event type
    ---------            ------           ----------
    04-15 05:44:43.992   SID-MGR(2)       ALLOC
```

Example 4: The following examples show how to view stale SIDs:

```
router# show segment-routing srv6 sid stale
SID                        Locator         Behavior             Context
       Owner
---                        -------         --------             -------

FC01:101:2:E002::           loc1           uA (PSP/USD)         Ethernet2/0 2001::99:2:3:3
```

```
router# show segment-routing srv6 sid stale detail
SID: FC01:101:2:E002::
 Behavior: uA (PSP/USD) (57)
 Context:
    interface: Ethernet2/0
    vrf: (not-set), v4-topo-id: 0xFFFF, v6-topo-id: 0xFFFF
    next-hop: 2001::99:2:3:3
    policy: (not-set)
    distinguisher: (not-set)
 Event history:
    Timestamp            Client           Event type
    ---------            ------           ----------
    04-15 06:58:13.961   router isis sr(3 ALLOC
    04-15 07:24:49.831   router isis sr(3 DEALLOC
```

Example 5: The following examples show how to view the configured IPv6 route and router prefix:

```
router# show ipv6 route
(snip)
C   FC01:101:2::/48 [0/0]
     via SR0, directly connected
L   FC01:101:2::/128 [0/0]
     via SR0, receive
I2  FC01:101:3::/48 [115/10]
     via FE80::A8BB:CCFF:FE02:8F02, Ethernet2/0
     via FE80::A8BB:CCFF:FE02:8F12, Ethernet2/1
I2  FC01:101:4::/48 [115/10]
     via FE80::A8BB:CCFF:FE01:C901, Ethernet3/0
```

```
     via FE80::A8BB:CCFF:FE01:C911, Ethernet3/1
I2  FC01:101:5::/48 [115/10]
     via FE80::A8BB:CCFF:FE03:A404, Ethernet4/0
     via FE80::A8BB:CCFF:FE03:A414, Ethernet4/1
```

```
router# show ipv6 route FC01:101:2::/48
Routing entry for FC01:101:2::/48
  Known via "connected", distance 0, metric 0, type connected
  Route count is 1/1, share count 0
  Routing paths:
    directly connected via SR0
      Route metric is 0, traffic share count is 1
      Last updated 00:37:54 ago
```

Example 6: This example shows how to view the configured express forwarding path (CEF):

```
router# show ipv6 cef FC01:101:2::/48  internal
FC01:101:2::/48, epoch 0, flags [att, cnn, srsid], RIB[C], refcnt 5, per-destination sharing

  sources: SRv6-SID, RIB
  feature space:
    IPRM: 0x00038004
    Broker: linked, distributed at 2nd priority
  subblocks:
    SRv6 SID: FC01:101:2::/48
     Block-len:32 Node-len:16 Func-len:0 Arg-len:0
     END Flags:0x1 OCE:
      End OCE stats:
        packet count:  0
        byte count:    0
        punt packet count:  0
        punt byte count:    0
        error count:   0
     SRv6 end 0x80007FF32CFA6F38, 4 locks [Flags: clean]
       Lookup in input interface's IPv6 table
  ifnums: (none)
  path list 7FF32C863280, 21 locks, per-destination, flags 0x65 [shble, hvsh, rcrsv, hwcn]
    path 7FF32C85D978, share 1/1, type recursive, for IPv6
      recursive via ::[IPv6:Default], fib 7FF32C87D000, 1 terminal fib, v6:Default:::/127
      path list 7FF32C8631D0, 2 locks, per-destination, flags 0x61 [shble, rcrsv, hwcn]
          path 7FF32C85D8A8, share 1/1, type recursive, for IPv6, flags [dsnt-src-via,
cef-intnl]
           recursive via ::/127<nh:::>[IPv6:Default], fib 7FF32C3592D8, 1 terminal fib,
v6:Default:::/127
           path list 7FF32C2F5860, 5 locks, per-destination, flags 0x41 [shble, hwcn]
               path 7FF32BF8ED50, share 1/1, type special prefix, for IPv6
                 discard
  output chain:
    SRv6 end 0x80007FF32CFA6F38, 5 locks [Flags: clean]
      Lookup in input interface's IPv6 table
```

SRv6 under IS-IS

SRv6 under IS-IS

Intermediate System-to-Intermediate System (IS-IS) protocol already supports segment routing with MPLS dataplane (SR-MPLS). From Cisco IOS XE 17.12.1a, IS-IS is extended to support Segment Routing with

IPv6 data plane (SRv6). The extensions include advertising the SRv6 capabilities of nodes, node segments, and adjacency segments as SRv6 SIDs.

Information About SRv6 under IS-IS

SRv6 under IS-IS performs the following functionalities:

- Interacts with SID Manager to learn local locator prefixes and announces the locator prefixes in the IGP domain.
- Learns remote locator prefixes from other IS-IS neighbor routers and installs the learned remote locator IPv6 prefix in RIB.
- Allocates or learns prefix SID and adjacency SIDs, creates local SID entries, and advertises them in the IGP domain.

Configuring SRv6 under IS-IS

Use the **segment-routing srv6** command under the **router isis** command to enable SRv6 under the IS-IS IPv6 address-family as shown in the examples below. Use the **level** {**1**| **2**} keywords to advertise the locator only in the specified IS-IS level.

For basic SRv6 configuration, see section Configuring SRv6.

The following example shows how to configure SRv6 under IS-IS.

```
Router(config)# router isis core
Router(config-isis)# address-family ipv6 unicast
Router(config-isis-af)# router-id Loopback0
Router(config-isis-af)# segment-routing srv6
Router(config-isis-af-srv6)# locator loc5
Router(config-isis-af-srv6-locator)# level 1
Router(config-isis-srv6-locator)# exit
```

The following example shows how to assign multiple SRv6 locators under IS-IS.

```
Router(config)# router isis core
Router(config-isis)# address-family ipv6 unicast
Router(config-isis-af)# segment-routing srv6
Router(config-isis-srv6)# locator myLocBestEffort
Router(config-isis-srv6-loc)# exit
Router(config-isis-srv6)# locator myLocLowLat
Router(config-isis-srv6-loc)# exit
```

For more information about configuring IS-IS, see chapter IS-IS Overview and Basic Configuration in the *Cisco IP Routing Configuration Guide*.

Note The **router-id** keyword enables the use of SRv6 policy.

Verifying SRv6 IS-IS Configuration

Example 1: Use the **show segment-routing srv6 locator** command to verify SRv6 under IS-IS configuration:

```
Router# show segment-routing srv6 locator

Name                    ID     Algo     Prefix              Status     Flags

myLoc1                  3       0       2001:0:8::/48         Up         U
myLocBestEffort         5       0       2001:0:1::/48         Up         U
```

Example 2: Use the **show isis srv6 locators** command to view SID locators.

```
router# show isis srv6 locators
ISIS SRv6 Locators:
Tag sr:
Name              Prefix                        Level
----              ------                        -----
loc1              FC01:101:2::/48               2
```

```
router# show isis srv6 locators detail
ISIS SRv6 Locators:
Tag sr:
Name              Prefix                         Level
----              ------                         -----
loc1              FC01:101:2::/48                2

Level-1 metric: 0
 Level-2 metric: 0
 End-SIDs:
   FC01:101:2::
```

SRv6 BGP-Based Services

SRv6 BGP-Based Services

Feature Name	Release	Description
Dual-Stack L3VPN Services (IPv4, IPv6) (SRv6 Micro-SID)	Cisco IOS XE Release 17.12.1a	This feature introduces support SRV6 for VPNV4 and VPNv6 VRFs. uDT4 and uDT6 based SRv6 service on the same interface, subinterface, or VRF are supported.

Building on the messages and procedures defined in IETF draft BGP/MPLSIP Virtual Private Networks (VPNs), BGP has been extended to provide the following services over an SRv6 network:

- IPv4 Layer-3 VPNs
- IPv6 Layer-3 VPNs

Based on the messages and procedures defined in IETF draft SRv6 BGP based Overlay services, BGP encodes the SRv6 Service SID in the prefix-SID attribute of the corresponding BGP updates, and advertises it to its IPv6 BGP peers.

For more information about BGP, refer to the chapter Cisco BGP Overview in the *Cisco IP Routing Configuration Guide, Cisco IOS XE 17.x*.

Restrictions for SRv6 BGP-Based Services

- The following SRv6 BGP-based services are supported:
  - IPv4 L3VPN
  - IPv6 L3VPN
- uDT4, uDT6, and uDT46 for L3VPN are supported.
- BGP does not support uDT46 allocation and advertisement.

Information About SRv6 BGP-Based Services

SRv6 Locator Inheritance Rules

SRv6 locators can be assigned at different levels inside the BGP routing process. BGP allocates SRv6 Service SIDs from configured locator spaces according to the following inheritance rules:

1. Use the locator as defined under the service. If not defined under the specific service, then:
2. Use the locator as defined under the corresponding address-family. If not defined under the corresponding address-family, then:
3. Use the locator as defined globally under BGP.

There are multiple places under BGP where locator(s) are specified:

- Global (most generic)
- VPN AF
- VRF AF (most specific)

If there is no specific locater configured, then the locator config from upper level is inherited in the following order:

Global -> VPN-AF -> VRF-AF



Note There is no default SRv6 SID allocation mode, and Locator mode cannot be configured without SRv6 SID allocation mode. If there is no locator configured or inherited, then BGP does not allocate SIDs.

BGP Handling of SID Manager Locator Changes

In the event that BGP configured locator does not exist in the SID manager,

- BGP configuration is accepted but is not active.
- BGP generates a syslog.
- BGP listens to locator config notifications from SID manager.

In the event that BGP configured locator is created in the SID manager,

- BGP is notified by SID manager of the creation.
- BGP activates if any matching locator config.
- BGP allocates SIDs for the relevant prefixes and advertises them.

In the event that BGP configured locator is deleted from the SID manager,

- SID Manager notifies BGP of the deletion.
- BGP deactivates if any matching locator config.
- BGP deallocates SIDs for the relevant prefixes and withdraws them.

In the event that BGP configured locator is modified (i.e. locator prefix is modified) in the SID manager,

- SID Manager notifies BGP of the change.
- BGP release all SIDs associated with the previous locator prefix.
- BGP allocates new SIDs for the new locator prefix and advertise updated prefixes.

For more information on how to configure an SRv6 locator, see section ConfiguringSRv6.

SRv6 Based L3VPN

This section provides information about L3VPNs (VPNv4 and VPNv6) over an SRv6 network.

The following restrictions apply to L3VPNs over an SRv6 network:

- Only Per-VRF allocation mode is supported (uDT4 and uDT6 behavior).
- Equal-Cost Multi-path (ECMP) is supported; Unequal Cost Multipath (UCMP) is not supported.
- MPLS L3VPN and SRv6 L3VPN interworking gateway is not supported.

Configuring SRv6 based L3VPN

To enable SRv6-based L3VPN, you need to enable SRv6 under BGP, specify the locator, and configure the SID allocation mode. The assignment of the locator can be done in different places under the **router bgp** configuration. See section SRv6 Locator Inheritance Rules.

Enabling SRv6 Globally under BGP

Use the **segment-routing srv6** command under the **router bgp as-number** command to enable SRv6 globally under the BGP routing process. The *as-number* range is from 1-65535.

```
router bgp 65000
 segment-routing srv6
  locator loc1
 exit-srv6
 !
```

Configuring SRv6 IPv4 L3VPN

This example shows the complete configuration for SRv6 based IPv4 L3VPN.

```
router bgp 65000
 !
 bgp router-id interface Loopback1
 no bgp default ipv4-unicast
 neighbor 2001::1:1:1:4 remote-as 65000
 neighbor 2001::1:1:1:4 update-source Loopback1
 address-family vpnv4
  !
  segment-routing srv6
   locator loc1
   alloc-mode per-vrf
  exit-srv6
  !
  neighbor 2001::1:1:1:4 activate
  neighbor 2001::1:1:1:4 send-community both
```

Configuring SRv6 IPv6 L3VPN

This example shows the complete configuration for SRv6 based IPv6 L3VPN.

```
router bgp 65000
 !
 bgp router-id interface Loopback1
 no bgp default ipv4-unicast
 neighbor 2001::1:1:1:4 remote-as 65000
 neighbor 2001::1:1:1:4 update-source Loopback1
 address-family vpnv6
  !
  segment-routing srv6
   locator loc1
   alloc-mode per-vrf
  exit-srv6
  !
  neighbor 2001::1:1:1:4 activate
  neighbor 2001::1:1:1:4 send-community both
```

Configuring SRv6 IPvx VRF L3VPN

This example shows the complete configuration for SRv6 based L3VPN for address family IPvx VRF.

```
router bgp 65000
 !
 bgp router-id interface Loopback1
 no bgp default ipv4-unicast
 neighbor 2001::1:1:1:4 remote-as 65000
 neighbor 2001::1:1:1:4 update-source Loopback1
 address-family ipv4 vrf ce1
  !
  segment-routing srv6
   locator loc1
   alloc-mode per-vrf
  exit-srv6
  !
  neighbor 99.1.2.1 remote-as 65001
  neighbor 99.1.2.1 activate
  neighbor 99.1.2.1 send-community both
 address-family ipv6 vrf ce1
  !
  segment-routing srv6
   locator loc1
   alloc-mode per-vrf
  exit-srv6
```

```
!
neighbor 1002::1 remote-as 65002
neighbor 1002::1 activate
neighbor 1002::1 send-community both
```

SRv6 eBGP Inter-AS

From Cisco IOS XE 17.16.1a, the following BGP inter-autonomous systems (Inter-AS) functionalities are supported over SRv6:

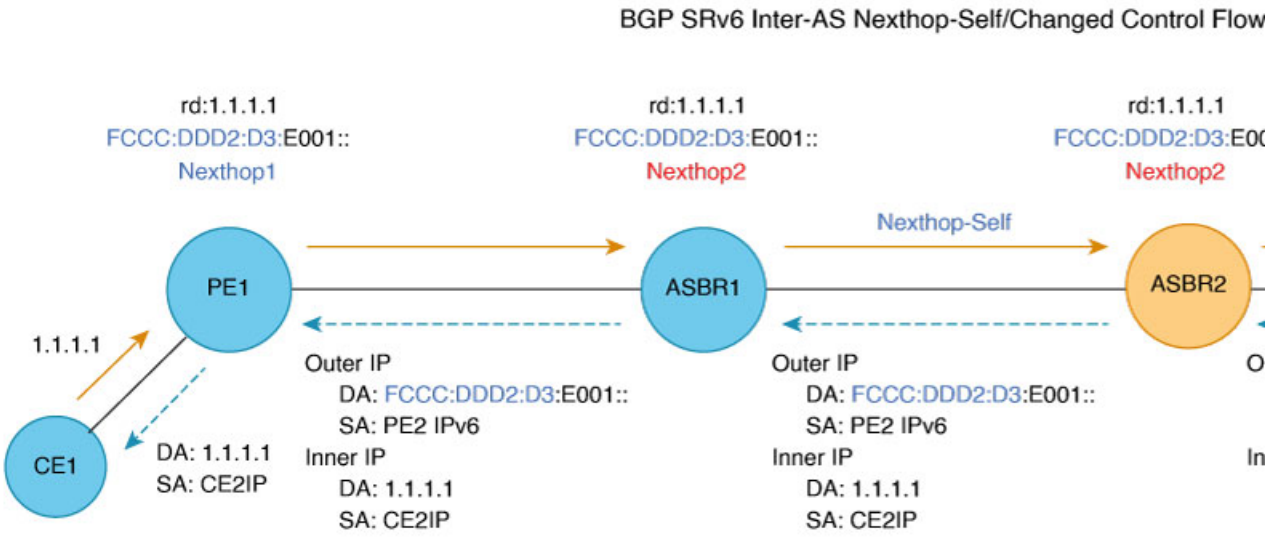
- BGP SRv6 Inter-AS Nexthop-Self/Changed
- BGP SRv6 Inter-AS Nexthop Unchanged

eBGP SRv6 Inter-AS Nexthop-Self/Changed

In this scenario, autonomous system boundary routers (ASBRs) propagate the BGP SRv6 L3VPN updates with SRv6 SID unchanged, but the next-hop destination changed. The reachability of the Egress PE to the SRv6 locator must be advertised across AS domains.

End-to-End Inter-AS SRv6 SRTE policy cannot be created by the headend node; it can only create SRTE policy for the ASBRs.

Figure 2: eBGP SRv6 Inter-AS Nexthop-Self/Changed

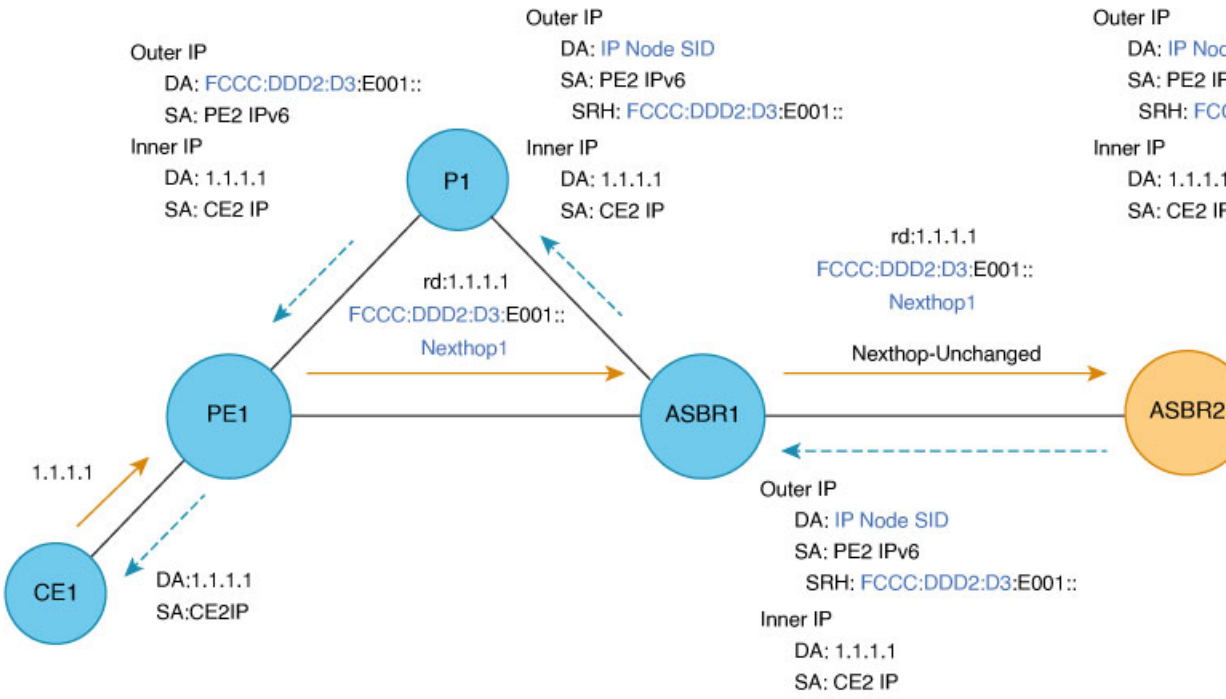


BGP SRv6 Inter-AS Nexthop Unchanged

In this scenario, ASBRs propagate the BGP SRv6 L3VPN updates with both SRv6 SID and next-hop unchanged. The reachability of the next-hop of the Egress PE to the SRv6 locator must be advertised across AS domains.

End-to-End Inter-AS SRv6 SRTE policy can be created by the headend node to the ingress endpoint if the complete network is managed by single IGP domain.

Figure 3: eBGP SRv6 Inter-AS Nexthop Unchanged



Configure SRv6 eBGP Inter-AS Nexthop Unchanged

The following sample configuration shows how to configure eBGP inter-AS with the SRv6 SID as well as the nexthop destination unchanged:

```
router bgp 100
  neighbor A001::1 disable-connected-check
  neighbor A001::1 remote-as 200
  neighbor A001::1 ebgp-multihop 255
  neighbor A001::1 update-source Loopback20
!
 address-family vpnv4
  neighbor A001::1 encap srv6
  neighbor A001::1 activate
  neighbor A001::1 send-community both
  neighbor A001::1 next-hop-unchanged
!
 address-family vpnv6
  neighbor A001::1 encap srv6
  neighbor A001::1 activate
  neighbor A001::1 send-community both
  neighbor A001::1 next-hop-unchanged
```

Configure SRv6 eBGP Inter-AS Nexthop Unchanged

The following sample configuration shows how to configure eBGP inter-AS with the SRv6 SID as well as the nexthop destination unchanged:

```
router bgp 100
  neighbor A001::1 disable-connected-check
  neighbor A001::1 remote-as 200
```

```
  neighbor A001::1 ebgp-multihop 255
  neighbor A001::1 update-source Loopback20
!
 address-family vpnv4
  neighbor A001::1 encap srv6
  neighbor A001::1 activate
  neighbor A001::1 send-community both
  neighbor A001::1 next-hop-unchanged
!
 address-family vpnv6
  neighbor A001::1 encap srv6
  neighbor A001::1 activate
  neighbor A001::1 send-community both
  neighbor A001::1 next-hop-unchanged
```

Verify SRv6 eBGP Inter-AS

Use the **show run** command to view the ASBR next-hop in an SRv6 eBGP configuration:

```
router#show run | se router bgp 100
router bgp 100
 neighbor A001::1 encap srv6
 neighbor A001::1 remote-as 200
 neighbor A001::1 ebgp-multihop 255
 neighbor A001::1 update-source Loopback20
!
 address-family vpnv4
  neighbor A001::1 encap srv6
  neighbor A001::1 activate
  neighbor A001::1 send-community both
  neighbor A001::1 next-hop-unchanged
```

```
router#show run | se router bgp 100
router bgp 100
 neighbor A001::1 encap srv6
 neighbor A001::1 remote-as 200
 neighbor A001::1 ebgp-multihop 255
 neighbor A001::1 update-source Loopback20
!
 address-family vpnv4
  neighbor A001::1 encap srv6
  neighbor A001::1 activate
  neighbor A001::1 send-community both
  neighbor A001::1 next-hop-self
```

BGP MPLS and SRv6 Co-Existence

A dual-connected PE that has both MPLS and SRv6 neighbors concurrently allocates a local MPLS label and an SRv6 SID for sourced/CE routes.

Restrictions

- MPLS label allocation is disabled when SRV6 is enabled for BGP AFI VRF.
- The **mpls alloc enable** command enables MPLS label allocation and is the default allocation mode. Both SRV6 and MPLS allocations are enabled, with MPLS being default allocation mode.
- MPLS label is advertised to a neighbor by default in the MPLS and SRv6 co-existence configuration.
- The **neighbor <> encap srv6** command is required to advertise SRv6 SID to a neighbor.

Configuring MPLS and SRv6 Coexistence for L3VPN

The following example shows the configuration to enable MPLS and SRv6 co-existence for L3VPN:

```
router bgp <instance>
    address-family [ipv4 | ipv6] unicast vrf <vrf-name>
        segment-routing srv6
                mpls alloc enable           >>>>> required for MPLS/SRv6 coexistence
    address-family vpnv4/vpnv6
                neighbor <A>                >>>>> can send any kind of update
                neighbor <B> encap srv6     >>>> SRv6 only neighbor
```



Note Sourced or CE prefixes from VRF's with MPLS and SRv6 coexistence enabled will be sent with MPLS labels.

Verifying SRv6 State

Use the following show commands to verify SRv6 BGP configurations.

Example 1: **`show segment-routing srv6 sid`**

```
device# show segment-routing srv6 sid
SID                   Locator     Behavior          Context                          Owner
---                   -------     --------          -------                          -----
FC01:101:2::          loc1        uN (PSP/USD)                                       SID-MGR

FC01:101:2:E000::     loc1        uDT4              ce1                              router
 bgp
FC01:101:2:E001::     loc1        uDT6              ce1                              router
 bgp
FC01:101:2:E002::     loc1        uA (PSP/USD)      Ethernet2/0 2001::99:2:3:3       router
 isis sr
FC01:101:2:E003::     loc1        uA (PSP/USD)      Ethernet2/1 2001::100:2:3:3      router
 isis sr
FC01:101:2:E004::     loc1        uA (PSP/USD)      Ethernet3/0 2001::99:2:4:4       router
 isis sr
FC01:101:2:E005::     loc1        uA (PSP/USD)      Ethernet3/1 2001::100:2:4:4      router
 isis sr
FC01:101:2:E006::     loc1        uA (PSP/USD)      Ethernet4/0 2001::99:2:5:5       router
 isis sr
FC01:101:2:E007::     loc1        uA (PSP/USD)      Ethernet4/1 2001::100:2:5:5      router
 isis sr
```

Example 2: **`show segment-routing srv6 sid <SID> detail`**

```
device# show segment-routing srv6 sid FC01:101:2:E000:: detail
SID: FC01:101:2:E000::   Type: DYNAMIC
 Behavior: uDT4 (63)
 Context:
    interface: (not-set)
    vrf: ce1, v4-topo-id: 0x1, v6-topo-id: 0xFFFF
    next-hop: (not-set)
    policy: (not-set)
    distinguisher: (not-set)
 Stats:
    Packets: 0  Bytes: 0
 User list:
    User:Refcount        Locator:Refcount
```

```
    -------------        -----------------
    router bgp(5):1      loc1:1
 Event history:
    Timestamp            Client           Event type
    ---------            ------           ----------
    04-15 07:24:08.165   router bgp(5)    ALLOC
```

Example 3: `show ip bgp all srv6 locator`

```
device# show ip bgp all srv6 locator
 Locator-1
   Name:  loc1
   Active:  Yes
   Refcount:  3
```

Example 4: `show ip bgp all srv6 sid`

```
device# show ip bgp all srv6 sid
 SID-1
   locator : loc1
   alloc-mode : 0
   status : ALLOCATED
   state : 1
   ref_count : 5
   topoid : 0x1E000001
   sid_value : FC01:101:2:E001::
   prefix_length : 64
   block_length : 32
   node _length : 16
   function _length : 16
   arg_length : 0
   behaviour : 62
 SID-2
   locator : loc1
   alloc-mode : 0
   status : ALLOCATED
   state : 1
   ref_count : 5
   topoid : 0x1
   sid_value : FC01:101:2:E000::
   prefix_length : 64
   block_length : 32
   node _length : 16
   function _length : 16
   arg_length : 0
   behaviour : 63
```

Example 5: `show ipv6 cef <prefix> internal`

```
device# show ipv6 cef FC01:101:8:E006:: internal
FC01:101:8:E006::/128, epoch 0, flags [att, srsid], refcnt 4, per-destination sharing
  sources: SRv6-SID
  subblocks:
    SRv6 SID: FC01:101:8:E006::/128
     Block-len:32 Node-len:16 Func-len:16 Arg-len:0
     END-DT4 Flags:0x5 OCE:
      End OCE stats:
        packet count:  20
        byte count:    2280
        punt packet count:  0
        punt byte count:    0
        error count:   0
```

```
     SRv6 end 0x80007FD05D9BC970, 4 locks [Flags: clean decap]
       Lookup in table IPv4:ce2
  ifnums: (none)
  path list 7FD05BD3F530, 21 locks, per-destination, flags 0x65 [shble, hvsh, rcrsv, hwcn]
    path 7FD05BD2D578, share 1/1, type recursive, for IPv6
      recursive via ::[IPv6:Default], fib 7FD05BD43C60, 1 terminal fib, v6:Default:::/127
      path list 7FD05BD3F480, 2 locks, per-destination, flags 0x61 [shble, rcrsv, hwcn]
          path 7FD05BD2D4A8, share 1/1, type recursive, for IPv6, flags [dsnt-src-via,
cef-intnl]
            recursive via ::/127<nh:::>[IPv6:Default], fib 7FD056DAB760, 1 terminal fib,
v6:Default:::/127
            path list 7FD054328EF8, 5 locks, per-destination, flags 0x41 [shble, hwcn]
                path 7FD05AF52578, share 1/1, type special prefix, for IPv6
                  discard
  output chain:
    SRv6 end 0x80007FD05D9BC970, 5 locks [Flags: clean decap]
      Lookup in table IPv4:ce2
```

Example 6: `show isis database verbose`

```
device# show isis database verbose
pe3.00-00          0x00000025   0xEF58                742/1198      0/0/0
  Area Address: 49
  NLPID:        0xCC 0x8E
  Topology:     IPv4 (0x0)
                IPv6 (0x2)
  Router ID:    1.1.1.8
  Router CAP:   1.1.1.8, D:0, S:0
    SRv6 Oflag:0
    Segment Routing: I:1 V:0, SRGB Base: 16000 Range: 8000
    Segment Routing Algorithms: SPF, Strict-SPF
    Segment Routing Local Block: SRLB Base: 30000 Range: 10000
    Node-MSD
      MSD: 16
  Hostname: iolpe3
(snip)
 SRv6 Locator: (MT-IPv6) FC01:101:8::/48 Metric:0 Algorithm:0
    End SID: FC01:101:8:: uN (PSP/USD)
      SID Structure:
        Block Length: 32, Node-ID Length: 16, Func-Length: 0, Args-Length: 0
```

Example 7: `show ipv6 route <prefix>`

```
device# show ipv6 route FC01:101:8::/48
Routing entry for FC01:101:8::/48
  Known via "isis sr", distance 115, metric 30, type level-2
  Route count is 4/4, share count 0
  Routing paths:
    FE80::A8BB:CCFF:FE01:E411, Ethernet3/1
      Route metric is 30, traffic share count is 1
      From FE80::A8BB:CCFF:FE01:E411
      Last updated 01:03:27 ago
    FE80::A8BB:CCFF:FE03:F504, Ethernet4/0
      Route metric is 30, traffic share count is 1
      From FE80::A8BB:CCFF:FE03:F504
      Last updated 01:03:27 ago
    FE80::A8BB:CCFF:FE03:F514, Ethernet4/1
      Route metric is 30, traffic share count is 1
      From FE80::A8BB:CCFF:FE03:F514
      Last updated 01:03:27 ago
    FE80::A8BB:CCFF:FE01:E401, Ethernet3/0
      Route metric is 30, traffic share count is 1
```

```
      From FE80::A8BB:CCFF:FE01:E401
      Last updated 01:03:27 ago
```

Example 8: **show bgp [vpnv4|vpnv6] rd <*rd*> <*prefix*>**

Sample output for VPNv4:

```
device# show bgp vpnv4 uni rd 1:1 22.22.22.22

BGP routing table entry for 1:1:22.22.22.22/32, version 13
Paths: (1 available, best #1, table red)
Not advertised to any peer
Refresh Epoch 1
3, imported path from 2:2:22.22.22.22/32 (global)
2023:1::1 (via default) from 1.1.1.3 (1.1.1.3)
Origin incomplete, metric 0, localpref 100, valid, internal, best
Extended Community: RT:1:1 RT:2:2
Originator: 11.1.1.1, Cluster list: 1.1.1.3
srv6 out-sid: FCCC:CCC1:AA88:E000::
rx pathid: 0, tx pathid: 0x0
Updated on Jun 28 2023 11:29:52 PST
```

Sample output for VPNv6:

```
device# show bgp vpnv6 uni rd 1:1 2222::1/128

BGP routing table entry for [1:1]2222::1/128, version 11
Paths: (1 available, best #1, table red)
Not advertised to any peer
Refresh Epoch 1
3, imported path from [2:2]2222::1/128 (global)
2023:1::1 (via default) from 1.1.1.3 (1.1.1.3)
Origin incomplete, metric 0, localpref 100, valid, internal, best
Extended Community: RT:2:2
Originator: 11.1.1.1, Cluster list: 1.1.1.3
srv6 out-sid: FCCC:CCC1:AA88:E001::
rx pathid: 0, tx pathid: 0x0
Updated on Jun 28 2023 11:29:52 PST
```

Example 9: **show ip route vrf <*vrf*> <*prefix*>**

```
device# show ip route vrf ce1 1.1.1.10
Routing Table: ce1
Routing entry for 1.1.1.10/32
  Known via "bgp 65000", distance 200, metric 0
  Tag 65010, type internal
  Last update from FC01:101:8:E006:: 08:51:34 ago
  Routing Descriptor Blocks:
  * FC01:101:8:E006:: (default:ipv6), from 1.1.1.4, 08:51:34 ago
      opaque_ptr 0x7FF32E0B9640
      Route metric is 0, traffic share count is 1
      AS Hops 1
      Route tag 65010
      MPLS label: none
```

Example 10: **show ipv6 route vrf <*vrf*> <*prefix*>**

```
device# show ipv6 route vrf red 2222::1/128

Routing entry for 2222::1/128
Known via "bgp 1", distance 200, metric 0
Tag 3, type internal
```

```
Route count is 1/1, share count 0
Routing paths:
FCCC:CCC1:AA88:E001::%default
Route metric is 0, traffic share count is 1
From ::FFFF:1.1.1.3
opaque_ptr 0x7FF38CDB6848
Last updated 00:03:16 ago
```

Example 11: **`show ip cef vrf <vrf> <prefix> internal`**

```
device# show ip cef vrf red 22.22.22.22 internal

22.22.22.22/32, epoch 0, flags [rnolbl, rlbls], RIB[B], refcnt 5, per-destination sharing
  sources: RIB
  feature space:
    IPRM: 0x00018000
  VPN-SID(s) on: 1/0:v4-rcrsv-FCCC:CCC1:AA88:E000::
  Path: v4-rcrsv-FCCC:CCC1:AA88:E000:: (VPN-SID: FCCC:CCC1:AA88:E000::)
    Flags: 00000004 [vpn-sid]
    IPv6 TC: 0   Hop Limit: 64
      Src: C02:1::7
      Dst: FCCC:CCC1:AA88:E000::
      Via: FCCC:CCC1:AA88:E000::
    Segment List (1)
      FCCC:CCC1:AA88:E000::
    Flow-based Encap Chains: 1
      IPV6 adj out of Ethernet0/0, addr FE80::A8BB:CCFF:FE00:3300 from FCCC:CCC1:AA88::/48
 <= SRv6 SID List OCE 0x7FF38D329078 (5) 1 Segments
  ifnums:
    Ethernet0/0(2): FE80::A8BB:CCFF:FE00:3300
  path list 7FF38CCDE0D8, 7 locks, per-destination, flags 0x8269 [shble, rif, rcrsv, hwcn,
 bgp, sb-oce]
    path 7FF38CCDB128, share 1/1, type recursive, for IPv4, flags [vpn-sid],
vpn-sid:FCCC:CCC1:AA88:E000::
      recursive via FCCC:CCC1:AA88:E000::[IPv6:Default], fib 7FF38CDA31B0, 1 terminal fib,
 v6:Default:FCCC:CCC1:AA88::/48
     path list 7FF38CCDDE18, 2 locks, per-destination, flags 0x69 [shble, rif, rcrsv, hwcn]

          path 7FF38CCDADE8, share 1/1, type recursive, for IPv6, flags [dsnt-src-via,
cef-intnl]
           recursive via FCCC:CCC1:AA88::/48<nh:FCCC:CCC1:AA88:E000::>[IPv6:Default], fib
 7FF38CDA3D78, 1 terminal fib, v6:Default:FCCC:CCC1:AA88::/48
           path list 7FF38CCDE658, 5 locks, per-destination, flags 0x49 [shble, rif, hwcn]

               path 7FF38CCDB7A8, share 1/1, type attached nexthop, for IPv6
               nexthop FE80::A8BB:CCFF:FE00:3300 Ethernet0/0, IPV6 adj out of Ethernet0/0,
 addr FE80::A8BB:CCFF:FE00:3300 7FF38CDE1848
  output chain:
    SRv6 SID List OCE 0x7FF38D329078 (8) 1 Segments
      Segment List (1)
        FCCC:CCC1:AA88:E000::
    PushCounter(SRv6 Encap) 7FF386CF0E58
    SRv6 Encap OCE 0x7FF38D328BE8 (4) fwd-id:0 FCCC:CCC1:AA88:E000::
      Flags: 00000004 [vpn-sid]
      IPv6 TC: 0   Hop Limit: 64
        Src: C02:1::7
        Dst: FCCC:CCC1:AA88:E000::
    IPV6 adj out of Ethernet0/0, addr FE80::A8BB:CCFF:FE00:3300 7FF38CDE1848
```

Example 12: **`show ipv6 cef vrf <vrf> <prefix> internal`**

```
device# show ipv6 cef vrf red 2222::1/128 internal

2222::1/128, epoch 0, RIB[B], refcnt 4, per-destination sharing
  sources: RIB
  feature space:
    IPRM: 0x00018000
  VPN-SID(s) on: 1/0:v6-rcrsv-FCCC:CCC1:AA88:E001::
  Path: v6-rcrsv-FCCC:CCC1:AA88:E001:: (VPN-SID: FCCC:CCC1:AA88:E001::)
    Flags: 00000004 [vpn-sid]
    IPv6 TC: 0   Hop Limit: 64
      Src: C02:1::7
      Dst: FCCC:CCC1:AA88:E001::
      Via: FCCC:CCC1:AA88:E001::
    Segment List (1)
      FCCC:CCC1:AA88:E001::
    Flow-based Encap Chains: 1
      IPV6 adj out of Ethernet0/0, addr FE80::A8BB:CCFF:FE00:3300 from FCCC:CCC1:AA88::/48
 <= SRv6 SID List OCE 0x7FF38D329018 (6) 1 Segments
  ifnums:
    Ethernet0/0(2): FE80::A8BB:CCFF:FE00:3300
  path list 7FF38CCDDD68, 9 locks, per-destination, flags 0x8269 [shble, rif, rcrsv, hwcn,
 bgp, sb-oce]
    path 7FF38CCDAD18, share 1/1, type recursive, for IPv6, flags [vpn-sid],
vpn-sid:FCCC:CCC1:AA88:E001::
      recursive via FCCC:CCC1:AA88:E001::[IPv6:Default], fib 7FF38CDA2E10, 1 terminal fib,
 v6:Default:FCCC:CCC1:AA88::/48
     path list 7FF38CCDDCB8, 2 locks, per-destination, flags 0x69 [shble, rif, rcrsv, hwcn]

          path 7FF38CCDAC48, share 1/1, type recursive, for IPv6, flags [dsnt-src-via,
cef-intnl]
           recursive via FCCC:CCC1:AA88::/48<nh:FCCC:CCC1:AA88:E001::>[IPv6:Default], fib
 7FF38CDA3D78, 1 terminal fib, v6:Default:FCCC:CCC1:AA88::/48
           path list 7FF38CCDE658, 5 locks, per-destination, flags 0x49 [shble, rif, hwcn]

                path 7FF38CCDB7A8, share 1/1, type attached nexthop, for IPv6
                nexthop FE80::A8BB:CCFF:FE00:3300 Ethernet0/0, IPV6 adj out of Ethernet0/0,
 addr FE80::A8BB:CCFF:FE00:3300 7FF38CDE1848
  output chain:
    SRv6 SID List OCE 0x7FF38D329018 (9) 1 Segments
      Segment List (1)
        FCCC:CCC1:AA88:E001::
    PushCounter(SRv6 Encap) 7FF386CF0DC8
    SRv6 Encap OCE 0x7FF38D328B48 (4) fwd-id:0 FCCC:CCC1:AA88:E001::
      Flags: 00000004 [vpn-sid]
      IPv6 TC: 0   Hop Limit: 64
        Src: C02:1::7
        Dst: FCCC:CCC1:AA88:E001::
    IPV6 adj out of Ethernet0/0, addr FE80::A8BB:CCFF:FE00:3300 7FF38CDE1848
device#
```

Troubleshooting and Debugging SRv6 BGP

The following BGP commands can be used to debug BGP updates:

- **debug bgp <> updates**
- **debug bgp <> addpath**

The following new command is introduced for debugging events related to SRv6:

- **debug ip bgp srv6**

EVPN VPWS over SRv6

Feature history for EVPN VPWS over SRv6

Table 2: Feature history

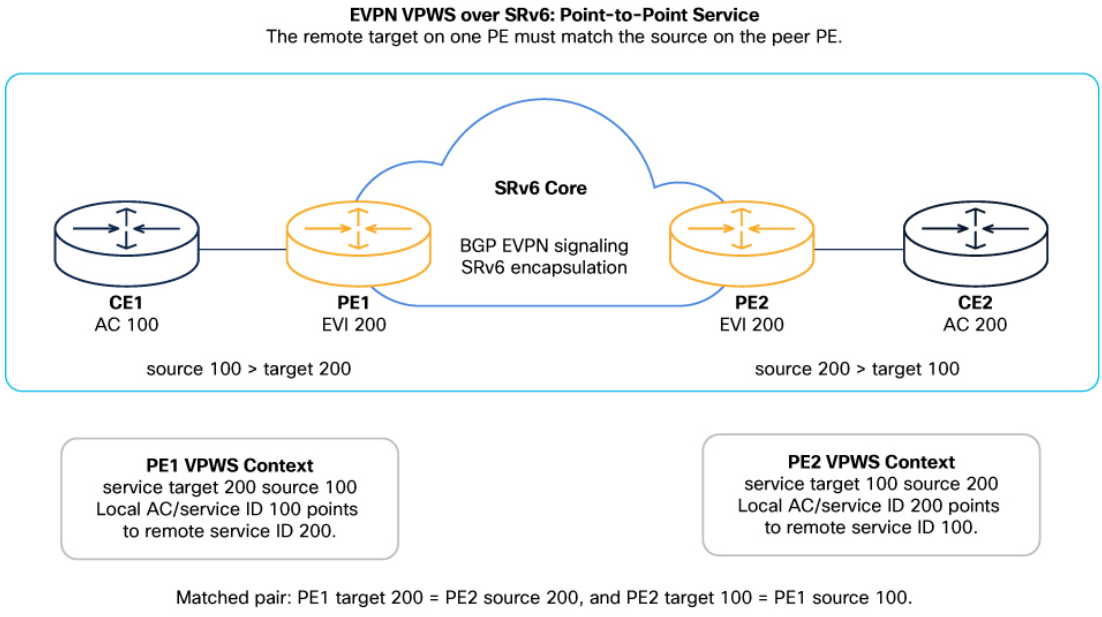
Feature Name	Release	Feature Information
EVPN VPWS over SRv6	Cisco IOS XE 26.2.1	This feature adds support for EVPN VPWS over SRv6 transport. EVPN VPWS uses EVPN signalling and SRv6 encapsulation to provide point-to-point Layer 2 VPN service between provider edge devices.

EVPN VPWS over SRv6

EVPN VPWS over SRv6 provides point-to-point Layer 2 VPN service using EVPN signalling and SRv6 encapsulation.

- EVPN VPWS forwards customer traffic to and from a customer attachment circuit without MAC lookup.
- EVPN VPWS establishes an EVPN instance between a pair of provider edge devices.
- The service uses BGP EVPN signalling and SRv6 encapsulation to carry VPWS traffic across the SRv6 transport.

Figure 4: EVPN VPWS over SRv6 point-to-point service



Supported platforms

- Cisco ASR1000
- Cisco Catalyst 8000 series edge
- Cisco Catalyst 8000V Edge Software
- Cisco 8000 Series Secure Routers

Restrictions for EVPN VPWS over SRv6

- Multihoming is not supported.
- Ethernet OAM interworking with EVPN VPWS over SRv6 is not upported.
- Remote link failure notification via EVPN VPWS over SRv6 is not supported on these platforms:
  - Cisco Catalyst 8500 Series Edge Platforms: C8500L-8S4X
  - Cisco Catalyst 8300 Series Edge Platforms: C8300-2N2S-4T2X
  - Cisco 1000 Series Integrated Services Routers: C1131X
  - 8100 Series Secure Routers: C8130-G2
  - 8200 Series Secure Routers
  - 8300 Series Secure Routers
  - 8400 Series Secure Routers: C8475-G2

Configure EVPN VPWS over SRv6

Perform the following steps to configure EVPN VPWS over SRv6:

Procedure

Step 1 Configure SRv6 with a locator.

```
segment-routing srv6
locators
  locator locator1
   prefix 2001:1:1::/48
   format usid-f3216
```

Step 2 Configure IGP with SRv6.

Configure IS-IS (or another IGP) to advertise SRv6 locator prefixes for IPv6 reachability.

Note: Multiple locators can be assigned to the IGP for advertisement.

```
router isis isis-1
 net 49.0001.0000.0000.0001.00
 is-type level-2-only
 router-id Loopback0
 metric-style wide
```

```
 !
 address-family ipv6
  segment-routing srv6
   locator locator2
    level-2
exit-address-family
```

Step 3 Configure BGP for EVPN-VPWS over SRv6.

Configure BGP with the L2VPN EVPN address family and enable SRv6 encapsulation for the EVPN neighbor. For eBGP EVPN neighbors, neighbor <address> encap srv6 is required to advertise EVPN SRv6 updates. For iBGP EVPN neighbors, this command is optional.

```
router bgp 1001
 bgp router-id interface Loopback0
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 2001:101:4:4::4 remote-as 1001
 neighbor 2001:101:4:4::4 update-source Loopback0
 !
 address-family l2vpn evpn
  neighbor 2001:101:4:4::4 activate
  neighbor 2001:101:4:4::4 send-community both
  neighbor 2001:101:4:4::4 encap srv6
 exit-address-family
```

Step 4 Configuring the EVPN VPWS Global SRv6 Locator

The global SRv6 locator under l2vpn evpn is used as the default locator for EVPN VPWS instances that use SRv6 encapsulation, unless a more specific locator is configured at the instance or VPWS context level.

```
l2vpn evpn
 logging vpws vc-state
 router-id Loopback0
 segment-routing srv6 locator locator1
```

Step 5 Configuring the EVPN-VPWS Instance with SRv6 Encapsulation

The encapsulation srv6 command enables SRv6 transport for the EVPN VPWS instance. If encapsulation is not configured, MPLS is used as the default transport.

The SRv6 locator can be configured globally under l2vpn evpn, at the EVPN instance level, or under the VPWS context. The most specific locator takes precedence.

```
l2vpn evpn instance 1 point-to-point
 encapsulation srv6
 segment-routing srv6 locator locator1 (optional)
 vpws context vc_1
  service target 20000 source 10000
  member TenGigabitEthernet0/0/2 service-instance 1
  segment-routing srv6 locator locator1 (optional)
 !
 vpws context vc_2
  service target 20001 source 10001
  member TenGigabitEthernet0/0/2 service-instance 10
  segment-routing srv6 locator locator1 (optional)
 !
```

Step 6 Configuring the CE-Facing Interface

EVPN VPWS over SRv6 supports the following attachment circuit interface modes:

- Port mode, using the main interface

```
 ! Port mode / main interface
l2vpn evpn instance 200 point-to-point
 encapsulation srv6
 vpws context vc200
  service target 200 source 200
  member TenGigabitEthernet0/0/8
```

- Sub-interface mode

```
 ! Sub-interface mode
interface TenGigabitEthernet0/0/8.200
 encapsulation dot1q 200
!
l2vpn evpn instance 200 point-to-point
 encapsulation srv6
 vpws context vc200
  service target 200 source 200
  member TenGigabitEthernet0/0/8.200
```

- Ethernet Flow Point (EFP) mode, using an Ethernet service instance

```
 ! EFP mode
interface TenGigabitEthernet0/0/8
 service instance 200 ethernet
  encapsulation dot1q 200
!
l2vpn evpn instance 200 point-to-point
 encapsulation srv6
 vpws context vc200
  service target 200 source 200
  member TenGigabitEthernet0/0/8 service-instance 200
```

EVPN VPWS over SRv6 configuration examples

The following examples show running configurations on PE1, PE2 and RR in an EVPN-VPWS over SRv6 deployment.

Example: PE1 Configuration

```
! Loopback Interface
interface Loopback0
 ipv6 address 2001:101:1:1::1/128
 ipv6 router isis isis-1
! Core-Facing Interfaces
interface TenGigabitEthernet0/0/0
 ipv6 address 2001:101:2:1::1/64
 ipv6 router isis isis-1
 isis network point-to-point
! CE-Facing Interface
interface TenGigabitEthernet0/0/2
 no ip address
 service instance 1 ethernet
  encapsulation dot1q 101
 !
 service instance 10 ethernet
```

```
  encapsulation default
 !
! SRv6 Configuration
segment-routing srv6
 locators
  locator locator1
   prefix 2001:1:1::/48
   format usid-f3216
! IGP Configuration
router isis isis-1
 net 49.0001.0000.0000.0001.00
 is-type level-2-only
 router-id Loopback0
 metric-style wide
 !
 address-family ipv6
  segment-routing srv6
   locator locator1
    level-2
 exit-address-family
! BGP Configuration
router bgp 1001
 bgp router-id interface Loopback0
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 2001:101:4:4::4 remote-as 1001
 neighbor 2001:101:4:4::4 update-source Loopback0
 !
 address-family l2vpn evpn
  neighbor 2001:101:4:4::4 activate
  neighbor 2001:101:4:4::4 send-community both
  neighbor 2001:101:4:4::4 encap srv6
 exit-address-family
! EVPN VPWS Configuration
l2vpn evpn
 logging vpws vc-state
 router-id Loopback0
 segment-routing srv6 locator locator1
l2vpn evpn instance 1 point-to-point
 encapsulation srv6
 vpws context vc_1
  service target 20000 source 10000
  member TenGigabitEthernet0/0/2 service-instance 1
 !
 vpws context vc_2
  service target 20001 source 10001
  member TenGigabitEthernet0/0/2 service-instance 10
 !
```

Example: PE2 Configuration

PE2 configuration is symmetric to PE1.

The service target and source identifiers are reversed (PE2's source matches PE1's target and vice versa).

The SRv6 locator prefix and loopback addresses are unique to PE2.

The BGP neighbor address points to PE1 (or the route reflector).

```
! Loopback Interface
interface Loopback0
 ipv6 address 2001:101:6:6::6/128
 ipv6 router isis isis-1
! Core-Facing Interfaces
```

```
interface TenGigabitEthernet0/0/0
 ipv6 address 2001:101:5:6::6/64
 ipv6 router isis isis-1
 isis network point-to-point
! CE-Facing Interface
interface TenGigabitEthernet0/0/5
 no ip address
 service instance 1 ethernet
  encapsulation dot1q 101
 !
 service instance 10 ethernet
  encapsulation default
 !
! SRv6 Configuration
segment-routing srv6
 locators
  locator locator1
   prefix 2001:1:6::/48
   format usid-f3216
! IGP Configuration
router isis isis-1
 net 49.0001.0000.0000.0006.00
 is-type level-2-only
 router-id Loopback0
 metric-style wide
 !
 address-family ipv6
  segment-routing srv6
   locator locator1
    level-2
 exit-address-family
! EVPN VPWS Configuration
l2vpn evpn
 logging vpws vc-state
 router-id Loopback0
 segment-routing srv6 locator locator1
l2vpn evpn instance 1 point-to-point
 encapsulation srv6
 vpws context vc_1
  service target 10000 source 20000
  member TenGigabitEthernet0/0/5 service-instance 1
 !
 vpws context vc_2
  service target 10001 source 20001
  member TenGigabitEthernet0/0/5 service-instance 10
 !
! BGP Configuration
router bgp 1001
 bgp router-id interface Loopback0
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 2001:101:4:4::4 remote-as 1001
 neighbor 2001:101:4:4::4 update-source Loopback0
 !
 address-family l2vpn evpn
  neighbor 2001:101:4:4::4 activate
  neighbor 2001:101:4:4::4 send-community both
  neighbor 2001:101:4:4::4 encap srv6
 exit-address-family
Important: Ensure the service identifiers are symmetric across PEs. PE1's source 10000 /
target 20000 must match PE2's source 20000 / target 10000.
```

Example: RR Configuration

```
interface Loopback0
 ipv6 address 2001:101:4:4::4/128
 ipv6 router isis isis-1
! SRv6 Configuration
segment-routing srv6
 locators
  locator locator1
   prefix 2001:188:4::/48
   format usid-f3216
! IGP Configuration
router isis isis-1
 net 49.0001.0000.0000.0004.00
 is-type level-2-only
 router-id Loopback0
 metric-style wide
 !
 address-family ipv6
  segment-routing srv6
   locator locator1
    level-2
 exit-address-family
! BGP Configuration
router bgp 1001
 bgp router-id interface Loopback0
 bgp log-neighbor-changes
 bgp graceful-restart
 no bgp default ipv4-unicast
 neighbor 2001:101:1:1::1 remote-as 1001
 neighbor 2001:101:1:1::1 log-neighbor-changes
 neighbor 2001:101:1:1::1 update-source Loopback0
 neighbor 2001:101:6:6::6 remote-as 1001
 neighbor 2001:101:6:6::6 log-neighbor-changes
 neighbor 2001:101:6:6::6 update-source Loopback0
 !
address-family l2vpn evpn
  neighbor 2001:101:1:1::1 activate
  neighbor 2001:101:1:1::1 send-community both
  neighbor 2001:101:1:1::1 route-reflector-client
  neighbor 2001:101:1:1::1 encap srv6
  neighbor 2001:101:6:6::6 activate
  neighbor 2001:101:6:6::6 send-community both
  neighbor 2001:101:6:6::6 route-reflector-client
  neighbor 2001:101:6:6::6 encap srv6
 exit-address-family
```

Verify EVPN VPWS over SRv6

Verifying SRv6 Locators

Use the show segment-routing srv6 locator command to verify that the configured SRv6 locators are in the Up state.

```
Device# show segment-routing srv6 locator
Name            Algo Prefix                                 Format              Status
----            ---- ------                                 ------              ------
locator1        0    2001:1:1::/48                          usid-f3216          Up
```

Verifying SRv6 SIDs

Use the show segment-routing srv6 sid command to verify SRv6 SID allocations. Look for uDX2 entries owned by evpn-srv6, which represent the local EVPN VPWS SIDs.

```
Device# show segment-routing srv6 sid
SID                                 Locator         Behavior
Context                                 Owner
---                                 -------         --------
-------                                 -----
2001:1:1::                          locator1        uN
(PSP/USD)                                                   SID-MGR
2001:1:1:E000::                   locator1       uDX2 (shift)         evi:1 eth_tag:10000
 if:Te0/0/2 efp:1     evpn-srv6
2001:1:1:E001::                   locator1       uDX2 (shift)         evi:1 eth_tag:10001
 if:Te0/0/2 efp:10    evpn-srv6
```

The uDX2 behavior indicates the Endpoint with decapsulation and Layer 2 cross-connect function. The context shows the EVI, Ethernet tag, interface, and EFP associated with each SID.

Verifying BGP EVPN Routes

Use the show bgp l2vpn evpn command to view the EVPN routing table summary.

```
Device# show bgp l2vpn evpn
BGP table version is 7, local router ID is 101.1.1.1
     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 101.1.1.1:1
*>   [1][101.1.1.1:1][00000000000000000000][10000]/23
                      ::                                    32768 ?
*>   [1][101.1.1.1:1][00000000000000000000][10001]/23
                      ::                                    32768 ?
*>i  [1][101.1.1.1:1][00000000000000000000][20000]/23
                      2001:101:6:6::6          0    100      0 ?
*>i  [1][101.1.1.1:1][00000000000000000000][20001]/23
                      2001:101:6:6::6          0    100      0 ?
```

Use the show bgp l2vpn evpn detail command to view detailed route information including SRv6 SIDs.

```
Device# show bgp l2vpn evpn detail
Route Distinguisher: 101.1.1.1:1
BGP routing table entry for [1][101.1.1.1:1][00000000000000000000][10000]/23, version 2
  Paths: (1 available, best #1, table evi_1)
  Local
    :: (via default) from 0.0.0.0 (101.1.1.1)
      Origin incomplete, localpref 100, weight 32768, valid, sourced, local, best
      Extended Community: RT:1001:1
      L2 srv6 in-sid: 2001:1:1:E000:: (pl:64 bl:32 nl:16 fl:16 al:0 bh:65)
BGP routing table entry for [1][101.1.1.1:1][00000000000000000000][20000]/23, version 6
  Paths: (1 available, best #1, table evi_1)
  Local, imported path from [1][101.6.6.6:1][00000000000000000000][20000]/23 (global)
    2001:101:6:6::6 (metric 40) (via default) from 2001:101:4:4::4 (101.4.4.4)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Extended Community: RT:1001:1
      Originator: 101.6.6.6, Cluster list: 101.4.4.4
      L2 srv6 out-sid: 2001:1:6:E000:: (pl:64 bl:32 nl:16 fl:16 al:0 bh:65)
```

Key fields to verify:

L2 srv6 in-sid — The local SRv6 SID for the VPWS VC (on locally sourced routes).

L2 srv6 out-sid — The remote SRv6 SID for the VPWS VC (on imported routes).

Next Hop — Must be a valid IPv6 address of the remote PE.

Extended Community: RT — Route targets must match between PEs.

Verifying EVPN VPWS VC Status

Use the show l2vpn evpn vpws vc all command for a brief summary.

```
Device# show l2vpn evpn vpws vc all
EVPN ID Source   Target   Type   Name/Interface                  Status
------- -------- -------- ------ -------------------------------- ----------
1       10000    20000    p2p    vc_1                            up
                                 Te0/0/2:1                       up
1       10001    20001    p2p    vc_2                            up
                                 Te0/0/2:10                      up
```

Use the show l2vpn evpn vpws vc all detail command for detailed VC information including SRv6 SIDs and traffic statistics.

```
Device# show l2vpn evpn vpws vc all detail
EVPN name: vc_1, state: up, type: point-to-point
  EVPN ID: 1
  VPWS Service Instance ID: Source 10000, Target 20000
  Encapsulation: srv6
  SRv6 Locator: locator1
  SRv6 SIDs: Local 2001:1:1:E000::, Remote 2001:1:6:E000::
  Next Hop Address: 2001:101:6:6::6
  Associated member interface Te0/0/2 up, Te0/0/2:1 status is up
  Rx Counters
    266856 input transit packets, 44089591 bytes
    0 drops
  Tx Counters
    260690 output transit packets, 64988756 bytes
    0 drops
  5 VC FSM state transitions, Last 5 shown
    Prov: Idle -> Prov, Wed Mar 25 15:25:23.769 (1w2d ago)
    AdmUp: Prov -> LocWait, Wed Mar 25 15:26:25.483 (1w2d ago)
    LocUp: LocWait -> RemWait, Wed Mar 25 15:26:25.484 (1w2d ago)
    RemUp: RemWait -> Act, Wed Mar 25 15:27:48.627 (1w2d ago)
    DpUp: Act -> Est, Wed Mar 25 15:27:48.633 (1w2d ago)
```

Key fields to verify:

Encapsulation: srv6 — Confirms SRv6 transport is in use.

SRv6 Locator — Shows the locator in use for this VC.

SRv6 SIDs: Local / Remote — Displays the local and remote END.DX2 SIDs.

Next Hop Address — IPv6 address of the remote PE.

VC FSM state — Should reach Est (Established) when fully operational.

Verifying EVPN VPWS Summary

```
Device# show l2vpn evpn vpws summary
L2VPN EVPN VPWS:
  EVIs (point-to-point): 1
  Total VCs: 2
    2 up, 0 down, 0 admin-down, 0 hot-standby, 0 other
  BGP: ASN 1001, address-family l2vpn evpn configured
  Router ID: 101.1.1.1
```

Verifying L2Route EVPN EAD Routes

Use the show l2route evpn ead per-evi command to view the L2 routing information base for EAD per-EVI routes.

```
Device# show l2route evpn ead per-evi
EVI   ETag       Prod   ESI                      Next Hop(s)
----- ---------- ------ ------------------------ ----------------------------------------
1     20000      L2VPN  N/A                      S:2001:1:1:E000:: Te0/0/2:1
1     20000      BGP    N/A                      S:2001:1:6:E000:: 2001:101:6:6::6
1     20001      L2VPN  N/A                      S:2001:1:1:E001:: Te0/0/2:10
1     20001      BGP    N/A                      S:2001:1:6:E001:: 2001:101:6:6::6
```

Use the show l2route evpn ead per-evi detail command for detailed information including SID parameters.

```
Device# show l2route evpn ead per-evi detail
EVPN Instance:            1
Ethernet Tag:             20000
Producer Name:            L2VPN
ESI:                      N/A
Next Hop(s):              Te0/0/2:1
Local SID:                2001:1:1:E000::
                             PL:64  BL:32  NL:16  FL:16
                             AL:0   Static:No  Behaviour:65
EVPN Instance:            1
Ethernet Tag:             20000
Producer Name:            BGP
ESI:                      N/A
Next Hop(s):              2001:101:6:6::6
                          SID: 2001:1:6:E000::
                             PL:64  BL:32  NL:16  FL:16
                             AL:0   Static:No  Behaviour:65
```

Producer: L2VPN — Locally originated routes.

Producer: BGP — Remotely learned routes.

Behavior: 65 — Corresponds to END.DX2 with Next C-SID.

Verifying L2FIB EVPN VPWS Encap

Use the show l2fib evpn vpws encap command to verify the forwarding plane programming.

```
Device# show l2fib evpn vpws encap
Type                          : EVPN_VPWS
Reference Count               : 1
SW Handle                     : 0x7F2A27A68FA0
PathList ID                   : 1
PathList Type                 : VPWS
Ethernet Segment              : 0000.0000.0000.0000.0000
Path Count                    : 1
Paths                         : [None] evi:1 etag:20000 [2001:101:6:6::6]
OCE Type                      : SRv6
Forwarding ID                 : 0x1800023
Forwarding OCE                : 0x7F2A276ECF90
Forwarding OCE Type           : 0x4E (srv6 sid list)
next hop track flags          : PRIMARY REGISTERED RESOLVED
Created                       : 1w2d ago
```

Note: - The Forwarding ID is the label in CEF table

Key fields to verify:

OCE Type: SRv6 — Confirms SRv6 output chain is programmed.

next hop track flags: PRIMARY REGISTERED RESOLVED — Confirms the SRv6 underlay path is resolved.

Verifying L2FIB EVPN VPWS Encap

Use the show ip cef label table command to verify the forwarding plane programming.

```
Device#show ip cef label-table
Label               Next Hop             Interface
0                    no route
25165859             FE80::768F:C2FF:FE47:8E31       TenGigabitEthernet0/0/1
                     FE80::768F:C2FF:FE48:78C1       TenGigabitEthernet0/0/0
25165860             FE80::768F:C2FF:FE47:8E31       TenGigabitEthernet0/0/1
                     FE80::768F:C2FF:FE48:78C1       TenGigabitEthernet0/0/0
```

Note: The label is the forwarding id of the VPWS virtual circuit

Verifying SRv6 Platform Capabilities

This command displays the platform SRv6 capabilities, including support for END.DX2 with uSID and EVPN VPWS single homing.

```
Device#show srv6 capabilities-parameters
Platform Capabilities:
 SRv6:Yes
 PFP:Yes
 MVPN:Yes
 TILFA:Yes
 EVPN VPWS Single-Homing:Yes
 Transit-Node-PTB-Relay, Headend-Node-PTB-Relay
 Endpoint behaviors:
  uN (PSP/USD)
  uA (PSP/USD)
  uDT6
  uDT4
  uDT46
  uDX2
  End.DTMC4
 Headend behaviors:
  H.ENCAP.RED
 Encap Parameters:
  Max-SL :16 (H-ENCAP-MSD:16 SL-MSD:16, END-POP-MSD:16 END-ENCAP-MSD:16)
  Encap :Collapsed
  Hop-limit propagate :Yes
  Traffic-class propagate :Yes
  Path-MTU supported :static
  SRH mode :
   combine :Yes
   stack :No
   insert :Yes

Parameters in-use:
 Encap Parameters:
  Source Address: 2001:101:1:1::1, Loopback0 (Default)
  Hop-Limit: 64 (Default)
  Traffic-class: 0 (Default)
  MTU: not configured (Default)
```

BGP SRv6 L3VPN On-Demand Next-Hop

BGP SRv6L3VPN On-Demand Next-Hop

When redistributing routing information across domains, provisioning of multi-domain services (L3VPN) has its own complexity and scalability issues. The On-Demand Next-Hop (ODN) configuration allows BGP to dynamically create SR policies as a result of learning routes with an extended color community attribute. It then installs the replied multi-domain LSP for the duration of the service into the local forwarding information base (FIB).

This section describes how SRv6 Traffic Engineering (SRv6-TE) works with the On-Demand Next-hop (ODN) mechanism.

Prerequisites for BGP SRv6 L3VPN ODN

Refer to the SRv6 under IS-IS section before configuring BGP SRv6L3VPN On-Demand Next-Hop.

Information About BGP SRv6L3VPN ODN

The Segment Routing-Traffic Engineering (SR-TE) On-Demand Next-Hop (ODN) is a mechanism that allows the steering of traffic on a segment routing policy based on the attributes of the packets. Packets are classified using Cisco's enhanced Policy-based Routing (ePBR) and then marked with internal tags known as Forward Classes (FCs). A PFP or PDP is used to route the marked packets based on the mappings between an FC and its corresponding path. This means that the traffic is steered based on its ePBR markings and switched to the appropriate path based on the FC of the packet.

BGP Color Extended Community and VRF Prefix Coloring

In the SR-TE mechanism, the prefix that needs an SR-TE routing path is associated with a color-extended community (an attribute that assigns color to the prefixes). Currently, BGP has the capability to attach the color-extended community based only on the neighbor command routemap outbound configuration. To color the prefixes based on attributes such as Source-VRF, Destination-VRF, CE-neighbor, and Source protocol, the following ways of attaching color are introduced:

- VRF Export Coloring
- VRF Import Coloring
- Route Redistribution Coloring into BGP
- Neighbor In-bound Coloring

Route-map Additive Color Extended Community

New color-extended communities from route-map can be added to the existing color extended communities list present in the prefix attribute if the new color is not already in the list. To add the new color extended community to the existing list of color extended communities of the prefix instead of replacing the existing, the keyword **additive** is used with the **route-map set extcommunity color** command:

```
route-map SRTE-color-map permit
         set extcommunity color < 1-4294967295> [additive]
```

Note

- If the **route-map set extcommunity color** command is configured without the **additive** keyword, the new color-extended community from route-map will replace any exiting color communities present in the prefix attribute.
- The number of color-extended communities that can be added to a prefix is limited by the number of extended communities that can be added to a BGP update message.

BGP Receiving of Multiple Colors

If BGP receives an update with multiple color extended communities, it creates an SR policy only for the highest color value, and not for other color values, as defined by the draft-ietf-spring-segment-routing-policy-08.

If the SR policy of highest color is down or unavailable, the BGP path will still be the best path, but will use the optimal path as routing path (for example, SRv6 SID as routing path).

The **show ip bgp <prefix>** command includes the SR policy information for the highest color, as SR policy is only created for the highest color value. The state of the highest color SR policy decides whether the SR policy or the optimal path is used as the routing path.

Note Multiple color handling, as defined in latest drafts (draft-ietf-spring-segment-routing-policy-09, RFC 9256) to use SR Policy of a lower color value when the SR policy of higher color value is down or unavailable, is not supported.

Configuring SRv6 L3VPN ODN

The following examples show how to configure SRv6 L3VPN ODN:

PE1 (Egress) Configuration

```
route-map test permit 10
 set extcommunity color 10
!

segment-routing srv6
 locators
  locator foo
   prefix FCCC:CCC2:C3::/48
   format usid-f3216
 !
!

router bgp 1
  neighbor 2023:1::3 remote-as 1
 !
address-family vpnv4
  segment-routing srv6
   locator foo
   alloc-mode per-vrf
  exit-srv6
          !
  neighbor 2023:1::3 activate
  neighbor 2023:1::3 send-community both
```

```
  neighbor 2023:1::3 route-map test out
exit-address-family
!
```

PE2 (Ingress) Configuration

```
segment-routing traffic-eng
 on-demand color 10
  authorize
  candidate-paths
   preference 1
    constraints
     segments
      dataplane srv6
     !
    !
    dynamic

router bgp 1
  neighbor 2023:1::3 remote-as 1
 !
 address-family vpnv4
  neighbor 2023:1::3 activate
  neighbor 2023:1::3 send-community both
 exit-address-family

 address-family vpnv6
  neighbor 2023:1::3 activate
  neighbor 2023:1::3 send-community both
 exit-address-family
```

Configuring SRv6 ODN Color Template

For BGP to dynamically instantiate SR-TE SRv6 policies to steer traffic onto, on-demand next-hop (ODN) color templates are used to define the attributes of the policies. These templates already exist and are extended for SRv6 as shown in the following sample configuration:

```
ipv6 prefix-list From-PE6 seq 5 permit A006::1/128
!
segment-routing traffic-eng
 on-demand color 1000
  authorize restrict
   ipv6 prefix-list From-PE6
  candidate-paths
   preference 100
     per-flow
      forward-class 0 color 10000
      forward-class 1 color 10001
      forward-class 2 color 10002
      forward-class 3 color 10003
      forward-class 4 color 10004
  !
 on-demand color 10000
  authorize restrict
   ipv6 prefix-list From-PE6
  candidate-paths
   preference 100
    constraints
     segments
      dataplane srv6
     !
     affinity
      include-all
```

```
      name non-voice
     !
    !
   !
   dynamic
    metric
     type delay
    !
   !
  preference 50
   constraints
    segments
     dataplane srv6
    !
    affinity
     include-all
      name voice
     !
    !
   !
   dynamic
   !
  !
 !
 performance-measurement
  delay-measurement
   liveness-detection
    invalidation-action down
!
on-demand color 10001
 authorize restrict
  ipv6 prefix-list From-PE6
 candidate-paths
  preference 100
   constraints
    segments
     dataplane srv6
    !
    affinity
     include-all
      name non-voice
     !
    !
   !
   dynamic
    metric
     type delay
    !
   !
  preference 50
   constraints
    segments
     dataplane srv6
    !
    affinity
     include-all
      name voice
     !
    !
   !
   dynamic
   !
  !
 !
```

```
  performance-measurement
   delay-measurement
    liveness-detection
     invalidation-action down
```

BGP Neighbor Outbound Prefix Coloring

Use this configuration to attach a color-extended community to an outbound BGP prefix update:

```
route-map SRTE-color-map permit
      set extcommunity color <1-4294967295> [additive]

   router bgp <ASnum>
     address-family <AF>
          neighbor <address> route-map SRTE-color-map out
     exit-address-family
      !
     address-family <AF> vrf <vrfname>
          neighbor <address> route-map SRTE-color-map out
     exit-address-family
```

BGP Neighbor Inbound Prefix Coloring

Use this configuration to attach a color-extended community to an inbound BGP prefix update:

```
route-map SRTE-color-map permit
      set extcommunity color <1-4294967295> [additive]

    router bgp <ASnum>
      address-family <AF>
          neighbor <address> route-map SRTE-color-map in
      exit-address-family
      !
     address-family <AF> vrf <vrfname>
          neighbor <address> route-map SRTE-color-map in
     exit-address-family
```

BGP VRF Export Prefix Coloring

Use this configuration to attach a color extended community to the VPN prefix per the export route-map color-extended community associated with the VRF:

```
route-map SRTE-color-map permit
      set extcommunity color <1-4294967295> [additive]

   vrf def SRTE-VRF
       rd 1:1
      !
       address-family ipv4
           export map SRTE-color-map
       exit-address-family
       !
       address-family ipv6
           export map SRTE-color-map
       exit-address-family
```

BGP VRF Import Prefix Coloring

Use this configuration to attach a color extended community to the VPN prefix per the import route-map color-extended community associated with the VRF:

```
route-map SRTE-color-map permit
      set extcommunity color <1-4294967295> [additive]
```

```
   vrf def SRTE-VRF
       rd 1:1
      !
       address-family ipv4
          import map SRTE-color-map
       exit-address-family
      !
       address-family ipv6
          import map SRTE-color-map
       exit-address-family
```

BGP Route Redistribution Prefix Coloring

Use this configuration to attach a color-extended community to a BGP prefix redistributed to BGP:

```
route-map SRTE-color-map permit
      set extcommunity color <1-4294967295> [additive]

    router bgp <ASnum>
      address-family ipv4 vrf <vrf-name>
          redistribute <source-protocol> route-map SRTE-color-map
     or
          network <address> mask <network-mask> route-map SRTE-color-map
      exit-address-family
      !
      address-family ipv6 vrf <vrf-name>
          redistribute <source-protocol=> route-map SRTE-color-map
     or
          network <address>/masklen route-map SRTE-color-map
      exit-address-family
```

Verifying SRv6 L3VPN ODN Configuration

Sample Output for BGP

The **show prefix** commands display the color and binding SID associated with the BGP prefix path:

- `show bgp vpnv4 unicast vrf <vrfname> <prefix>`
- `show bgp vpnv6 unicast vrf <vrfname> <prefix>`

```
device#show bgp vpnv4 unicast vrf red 22.22.22.22
BGP routing table entry for 1:1:22.22.22.22/32, version 44
Paths: (1 available, best #1, table red)
  Advertised to update-groups:
     1
  Refresh Epoch 2
  3, imported path from 2:2:22.22.22.22/32 (global)
    2023:1::1 (via default) from 1.1.1.3 (1.1.1.3)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Extended Community: RT:1:1 RT:2:2 Color:10
      Originator: 11.1.1.1, Cluster list: 1.1.1.3
      binding SID: 16777218 (color - 10) (state - UP)
      srv6 out-sid: FCCC:CCC1:AA88:E000::
      rx pathid: 0, tx pathid: 0x0
      Updated on Mar 29 2023 12:38:45 PST

device#show bgp vpnv6 unicast vrf red 2222::1/128
BGP routing table entry for [1:1]2222::1/128, version 45
Paths: (1 available, best #1, table red)
  Advertised to update-groups:
```

```
    1
  Refresh Epoch 2
  3, imported path from [2:2]2222::1/128 (global)
    2023:1::1 (via default) from 1.1.1.3 (1.1.1.3)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Extended Community: RT:2:2 Color:11
      Originator: 11.1.1.1, Cluster list: 1.1.1.3
      binding SID: 16777220 (color - 11) (state - UP)
      srv6 out-sid: FCCC:CCC1:AA88:E001::
      rx pathid: 0, tx pathid: 0x0
      Updated on Mar 30 2023 15:58:52 PST
```

Sample Output for SRv6 ODN Policy

The **show segment-routing traffic-eng policy name <name>** command displays the SRv6 ODN policy information:

```
device#show segment-routing traffic-eng policy name *A006::1|1000

Name: *A006::1|1000 (Color: 1000 End-point: A006::1)
  Owners : BGP
  Status:
    Admin: up, Operational: up for 00:01:16 (since 11-23 09:57:11.624)
  Candidate-paths:
    Preference 100 (BGP):
      Per-flow Information (active):
        Forward       PDP     PDP     BSID RW
          Class      Color   Status   Status
        -------- ---------- -------   --------
               0      10000      up   Pending
               1      10001      up   Pending
               2      10002      up   Pending
               3      10003      up   Pending
               4      10004      up   Pending
      Default Forward Class: 0
  Attributes:
  IPv6 caps enabled

device#show segment-routing traffic-eng policy name *A006::1|10000

Name: *A006::1|10000 (Color: 10000 End-point: A006::1)
  Owners : BGP-PFP-AUTO
  Status:
    Admin: up, Operational: up for 00:02:23 (since 11-23 09:56:34.669)
  Candidate-paths:
    Preference 100 (BGP-PFP-AUTO):
      PM State: Up
      Constraints:
        Affinity:
          include-all:
           non-voice
      Dynamic (active)
        Metric Type: DELAY, Path Accumulated Metric: 120
          F:1:1:E002:: [Adjacency-SID]
          F:1:2:E002:: [Adjacency-SID]
          F:1:3:E002:: [Adjacency-SID]
    Preference 50 (BGP-PFP-AUTO):
      PM State: Up
      Constraints:
        Affinity:
          include-all:
           voice
      Dynamic (inactive)
        Metric Type: TE, Path Accumulated Metric: 30
```

```
          F:1:4:: [Node-SID]
          F:1:5:: [Node-SID]
          F:1:6:: [Node-SID]
  Attributes:
```

The **show segment-routing traffic-eng policy name <name> detail** command displays the SRv6 ODN policy details:

```
device#show segment-routing traffic-eng policy name *A006::1|1000 detail

Name: *A006::1|1000 (Color: 1000 End-point: A006::1)
 Owners : BGP
 Status:
   Admin: up, Operational: up for 00:01:20 (since 11-23 09:57:11.624)
 Candidate-paths:
   Preference 100 (BGP):
     Per-flow Information (active):
       Forward        PDP     PDP   BSID RW
         Class      Color   Status   Status
      -------- ---------- -------  --------
             0      10000      up   Pending
             1      10001      up   Pending
             2      10002      up   Pending
             3      10003      up   Pending
             4      10004      up   Pending
    Default Forward Class: 0
  Attributes:
  IPv6 caps enabled
  Forwarding-ID: 16778667
  Per owner configs:
    BGP
      Binding SID: not configured
  Stats:
    Packets: 0  Bytes: 0

  Event history:
   Timestamp             Client          Event type          Context: Value
   ---------             ------          ----------          -------: -----
   11-23 09:56:34.668    BGP             Policy created      Name: BGP
   11-23 09:56:34.668    BGP             Set colour          Colour: 1000
   11-23 09:56:34.668    BGP             Set end point       End-point: A006::1
   11-23 09:56:34.669    BGP             Set dynamic pce     Path option: per flow
   11-23 09:57:11.624    FH Resolution   Policy state UP     Status: PFP RESOLVED CP: 100
   11-23 09:57:11.706    FH Resolution   REOPT triggered     Status: REOPTIMIZED CP: 100
   11-23 09:57:11.711    FH Resolution   REOPT triggered     Status: REOPTIMIZED CP: 100
   11-23 09:57:12.208    FH Resolution   REOPT triggered     Status: REOPTIMIZED CP: 100

device#show segment-routing traffic-eng policy name *A006::1|10000 detail

Name: *A006::1|10000 (Color: 10000 End-point: A006::1)
  Owners : BGP-PFP-AUTO
  Status:
    Admin: up, Operational: up for 00:02:28 (since 11-23 09:56:34.669)
  Candidate-paths:
    Preference 100 (BGP-PFP-AUTO):
      PM State: Up
      Constraints:
        Affinity:
          include-all:
           non-voice
      Dynamic (active)
        Metric Type: DELAY, Path Accumulated Metric: 120
          F:1:1:E002:: [Adjacency-SID]
          F:1:2:E002:: [Adjacency-SID]
          F:1:3:E002:: [Adjacency-SID]
```

```
  Preference 50 (BGP-PFP-AUTO):
    PM State: Up
    Constraints:
      Affinity:
        include-all:
         voice
    Dynamic (inactive)
      Metric Type: TE, Path Accumulated Metric: 30
        F:1:4:: [Node-SID]
        F:1:5:: [Node-SID]
        F:1:6:: [Node-SID]
Attributes:
Forwarding-ID: 16778668
Per owner configs:
  BGP-PFP-AUTO
    Binding SID: not configured
    Performance-measurement:
      liveness-detection
      invalidation-action down
Stats:
  Packets: 0  Bytes: 0
PM profile: Not configured

Event history:
  Timestamp           Client          Event type          Context: Value
  ---------           ------          ----------          -------: -----
  11-23 09:56:34.669  BGP-PFP-AUTO    Policy created      Name: BGP-PFP-AUTO
  11-23 09:56:34.669  BGP-PFP-AUTO    Set colour          Colour: 10000
  11-23 09:56:34.670  BGP-PFP-AUTO    Set end point       End-point: A006::1
  11-23 09:56:34.670  BGP-PFP-AUTO    Set dynamic         Path option: dynamic
  11-23 09:56:34.672  BGP-PFP-AUTO    Set dynamic         Path option: dynamic
  11-23 09:56:34.673  BGP-PFP-AUTO    Set delay measure   status: Enabled
  11-23 09:56:34.673  BGP-PFP-AUTO    Set Live Detection  status: Enabled
  11-23 09:56:34.673  BGP-PFP-AUTO    Set Live Invalidatio action: down
  11-23 09:57:07.662  FH Resolution   Liveness            CP: 100, SL2153 is Waiting
  11-23 09:57:07.663  FH Resolution   Liveness            CP: 50, SL2154 is Waiting
  11-23 09:57:11.609  PM              Liveness            CP: 50, SL2154 is Up
  11-23 09:57:11.623  FH Resolution   Policy state UP     Status: PATH RESOLVED CP: 50
  11-23 09:57:12.740  PM              Liveness            CP: 100, SL2153 is Up
  11-23 09:57:12.766  FH Resolution   REOPT triggered     Status: REOPTIMIZED CP: 100
  11-23 09:57:18.334  FH Resolution   REOPT triggered     Status: REOPTIMIZED CP: 100
  11-23 09:57:18.382  PM              Liveness            CP: 50, SL2154 is Unknown
  11-23 09:57:19.922  FH Resolution   REOPT triggered     Status: REOPTIMIZED CP: 100
  11-23 09:57:28.707  FH Resolution   REOPT triggered     Status: REOPTIMIZED CP: 100
  11-23 09:57:28.708  FH Resolution   Liveness            CP: 50, SL2154 is Waiting
  11-23 09:57:28.709  FH Resolution   REOPT triggered     Status: REOPTIMIZED CP: 100
  11-23 09:57:32.000  PM              Liveness            CP: 50, SL2154 is Up
  11-23 09:57:32.003  FH Resolution   REOPT triggered     Status: REOPTIMIZED CP: 100
```

For more information about verifying SRv6-TE configuration, see section *Verifying SRv6-TE Configuration*.

Debugging SRv6 L3VPN ODN Configuration

Use the following debug command to track the events related to SR ODN:

- **debug ip bgp sr-policy**

```
*Apr 10 17:35:48.773: BGP(4): 2023:1::3 rcvd UPDATE w/ attr: nexthop 2023:1::1, origin ?,
localpref 100, metric 0, originator 11.1.1.1, clusterlist 1.1.1.3, merged path 3, AS_PATH
, extended community RT:1:1 RT:2:2 Color:10, PrefixSid attribute: SRV6 SID FCCC:CCC1:AA88::
*Apr 10 17:35:48.773: BGP(4): 2023:1::3 rcvd 2:2:22.22.22.22/32, label 2162163712 (0x80E00000)
*Apr 10 17:35:48.773: BGP SRv6 SID ATTR: blk 32 node 16 fun 16 arg 0 pos 16 off 48
*Apr 10 17:35:48.774: BGP-SR Policy (7F7911708510): Binding SID 10/2023:1::1/ request
```

```
*Apr 10 17:35:48.774: BGP(4): Revise route installing 1 of 1 routes for 22.22.22.22/32 ->
0.0.0.0(red) to red IP table

PE2#show bgp vpnv4 uni vrf red 22.22.22.22
BGP routing table entry for 1:1:22.22.22.22/32, version 33
Paths: (1 available, best #1, table red)
  Advertised to update-groups:
     3
  Refresh Epoch 1
  3, imported path from 2:2:22.22.22.22/32 (global)
    2023:1::1 (via default) from 1.1.1.3 (1.1.1.3)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Extended Community: RT:1:1 RT:2:2 Color:10
      Originator: 11.1.1.1, Cluster list: 1.1.1.3
      binding SID: 16777219 (color - 10) (state - UP)
      srv6 out-sid: FCCC:CCC1:AA88:E000::
      rx pathid: 0, tx pathid: 0x0
      Updated on Apr 10 2023 09:35:48 PST

PE2(config)#segment-routing traffic-eng
PE2(config-srte-on-demand-color)#on-demand color 10

PE2(config-srte-on-demand-color)#no authorize
*Apr 10 17:37:25.964: BGP SR: color change notification callback for color 10, auth_type 3
*Apr 10 17:37:25.964: BGP SR: color change state handler for color 10, color_state 2
*Apr 10 17:37:25.964: BGP SR: deletion of policy *2023:1::1|10 is successful
*Apr 10 17:37:25.964: BGP SR: sr_policymgr color 10, delete: timer started
*Apr 10 17:37:26.678: BGP SR: Policy change timer expired.: bgp_sr_policy_service_change
started
*Apr 10 17:37:26.679: BGP(4): Revise route installing 1 of 1 routes for 5.5.6.8/32 ->
0.0.0.0(red) to red IP table
*Apr 10 17:37:26.680: BGP(4): Revise route installing 1 of 1 routes for 10.1.1.0/24 ->
0.0.0.0(red) to red IP table
*Apr 10 17:37:26.680: BGP(4): Revise route installing 1 of 1 routes for 22.22.22.22/32 ->
0.0.0.0(red) to red IP table
PE2(config-srte-on-demand-color)#do sh bgp vpnv4 uni vrf red 22.22.22.22
BGP routing table entry for 1:1:22.22.22.22/32, version 30
Paths: (1 available, best #1, table red)
  Advertised to update-groups:
     3
  Refresh Epoch 1
  3, imported path from 2:2:22.22.22.22/32 (global)
    2023:1::1 (via default) from 1.1.1.3 (1.1.1.3)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Extended Community: RT:1:1 RT:2:2 Color:10
      Originator: 11.1.1.1, Cluster list: 1.1.1.3
      binding SID: none (color-10)
      srv6 out-sid: FCCC:CCC1:AA88:E000::
      rx pathid: 0, tx pathid: 0x0
      Updated on Apr 10 2023 09:35:48 PST


PE2(config-srte-on-demand-color)#authorize
*Apr 10 17:37:41.096: BGP SR: color change notification callback for color 10, auth_type 2
*Apr 10 17:37:41.096: BGP SR: color change state handler for color 10, color_state 1
*Apr 10 17:37:41.096: BGP SR POLICY: policy *2023:1::1|10 create request
*Apr 10 17:37:41.097: BGP SR: sr_policymgr color 10, add
*Apr 10 17:37:41.097: BGP SR: Policy change for *2023:1::1|10, type 5, bsid 0
*Apr 10 17:37:41.097: BGP SR Policy Change notification: color: 10, nexthop: 2023:1::1:
timer started
*Apr 10 17:37:41.097: BGP_SR_POLICY: policy *2023:1::1|10 registered for policy notification
*Apr 10 17:37:41.097: BGP SR: Policy change for *2023:1::1|10, type 3, bsid 1000003
*Apr 10 17:37:41.097: BGP SR Policy Change notification: color: 10, nexthop: 2023:1::1
*Apr 10 17:37:41.097: BGP SR Policy Found: color: 10, nexthop: 2023:1::1: timer already
```

```
running
*Apr 10 17:37:41.097: BGP SR: Policy change for *2023:1::1|10, type 4, bsid 1000003
*Apr 10 17:37:41.097: BGP SR Policy Change notification: color: 10, nexthop: 2023:1::1
*Apr 10 17:37:41.097: BGP SR Policy Found: color: 10, nexthop: 2023:1::1: timer already
running
*Apr 10 17:37:42.039: BGP SR: Policy change timer expired.: bgp_sr_policy_service_change
started
*Apr 10 17:37:42.040: BGP(4): Revise route installing 1 of 1 routes for 5.5.6.8/32 ->
0.0.0.0(red) to red IP table
*Apr 10 17:37:42.040: BGP(4): Revise route installing 1 of 1 routes for 10.1.1.0/24 ->
0.0.0.0(red) to red IP table
*Apr 10 17:37:42.040: BGP(4): Revise route installing 1 of 1 routes for 22.22.22.22/32 ->
0.0.0.0(red) to red IP table
PE2(config-srte-on-demand-color)#do sh bgp vpnv4 uni vrf red 22.22.22.22
BGP routing table entry for 1:1:22.22.22.22/32, version 33
Paths: (1 available, best #1, table red)
  Advertised to update-groups:
     3
  Refresh Epoch 1
  3, imported path from 2:2:22.22.22.22/32 (global)
    2023:1::1 (via default) from 1.1.1.3 (1.1.1.3)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Extended Community: RT:1:1 RT:2:2 Color:10
      Originator: 11.1.1.1, Cluster list: 1.1.1.3
      binding SID: 16777219 (color - 10) (state - UP)
      srv6 out-sid: FCCC:CCC1:AA88:E000::
      rx pathid: 0, tx pathid: 0x0
  Updated on Apr 10 2023 09:35:48 PST
```

For information about troubleshooting and debugging SRv6-TE, see the Troubleshooting and Debugging SRv6-TE section.

SRv6 Traffic Engineering Policies

SRv6 Traffic Engineering Policies

From Cisco IOS XE 17.12.1a, the Segment Routing Traffic Engineering (SR-TE) mechanism is extended to Segment Routing over IPv6 (SRv6).

Restrictions for SRv6-TE Policies

- From Cisco IOS XE 17.16.1a, local path computation and PCE delegated path computation are supported; PCE-initiated polices are not supported.
- Only dynamic segment-lists are supported; explicit segment-lists are not supported.
- SRv6 Binding SIDs are not supported.
- On-demand next-hop (ODN) is not supported.
- L2VPN over SR-TE is not supported.
- Auto-route announce over PFP or PDP is not supported.
- When you create a policy with multiple SIDs, the final SID to reach the egress PE is the node SID, and this will be removed from the SID list.
- The VPN SID must always have the locator information as part of the SIDs.

Information About SRv6-TE Policies

The SRv6 Traffic Engineering (SRv6-TE) uses a SRv6 policy to steer traffic through the network. The SRv6 policy includes Per-flow policies (PFP) and Per-destination policies (PDP), both of which are supported.

An ePBR policy is applied to the ingress interface to define how traffic is classified and associated with the forward-class (FC), PFP is configured with a Per-flow forward-class table up to eight entries. Each entry is indexed by a FC and points to a PDP.

For PFP, the packets are classified on the ingress interface and choose different PDP paths to forward to the same destination based on the classification by ePBR.

Configuring SRv6-TE

The following examples demonstrate how to configure locally computed SRv6-TE policies..

Configuring PDP

```
segment-routing traffic-eng
    policy SRV6PM
      color 1 end-point C02:1::1
      candidate-paths
       preference 1
        constraints
         segments
          dataplane srv6
         !
        !
        dynamic
        !
       !
      !
       preference 2
        constraints
         segments
          dataplane srv6
         !
         affinity
          exclude-any
           name blue
          !
         !
        !
        dynamic
           metric
              type delay
         !
        !
      performance-measurement
       delay-measurement
        liveness-detection
         invalidation-action down
        !
       !
      !
```

Configuring PFP

```
segment-routing traffic-eng
    policy PFP
      color 100 end-point C02:1::1
```

```
      candidate-paths
       preference 1
        per-flow
         forward-class 0 color 1
         forward-class 1 color 2
         forward-class 2 color 3
         forward-class 3 color 4
         forward-class 4 color 5
```

Configuring ePBR

```
policy-map type epbr PFP
 class FC1
  set forward-class 1
 class FC2
  set forward-class 2
 class FC3
  set forward-class 3
 class FC4
  set forward-class 4
 class class-default
  set forward-class 0

interface TenGigabitEthernet2/2/0.1000
 encapsulation dot1Q 1000
  vrf forwarding vpn-1000
   ip address 17.0.0.1 255.255.255.0
   ipv6 address 1700::1/64
   service-policy type epbr input PFP
```

Configuring Static Route

1. IPv6 static route for a prefix, NO SR policy, and optional VPN SID

   ```
   ipv6 route vrf blue 1002:1::/64 2001:1::2 nexthop-vrf default sid-list h-encaps-red
   FCCC:CCC1:C3:E005::
   ```

2. IPv6 static route for a prefix with traffic steered via optional SR policy and VPN SID

   ```
   ipv6 route vrf blue 1002:1::/64 segment-routing srv6 via policy PFP sid-list h-encaps-red
    FCCC:CCC1:C3:E005::
   ```

3. IPv4 static route for a prefix with traffic steered via optional SR policy and VPN SID

   ```
   ip route vrf blue 2.2.2.2 255.255.255.255 segment-routing srv6 via policy PFP sid-list
   h-encaps-red FCCC:CCC1:C3:E004::
   ```



Note IPv4 static route for a prefix, NO SR policy, and optional VPN SID is NOT supported.

Configuring PCE Delegation for Path Computation

The following sample configuration demonstrates how to change the locally computed SRv6-TE PDP in the previous example to be delegated for path computation to a PCE. From Cisco IOS XE 17.16.1a, you can configure PCE delegation for path computation.

```
segment-routing traffic-eng
    pcc
      pce address C5:1::1
    policy SRV6PM
```

```
     color 1 end-point C02:1::1
      candidate-paths
        preference 1
         constraints
           segments
            dataplane srv6
!
!
    dynamic
      pcep
!
!
!
        preference 2
         constraints
            segments
             dataplane srv6
!
    affinity
     exclude-any
      name blue
!
!
!
    dynamic
       pcep
        metric
         type delay
!
!
    performance-measurement
     delay-measurement
      liveness-detection
       invalidation-action down
!
!
!
```

Verifying SRv6-TE Configuration

Example 1: Use the **show segment-routing traffic-eng policy name** command to verify SRv6-TE configuration, with PDP and PFP:

```
router# show segment-routing traffic-eng policy name SRV6PM detail

Name: SRV6PM (Color: 1 End-point: C02:1::1)
  Owners : CLI
  Status:
    Admin: up, Operational: up for 70:55:04 (since 04-11 12:10:05.054)
  Candidate-paths:
    Preference 2 (CLI):
      PM State: Up
      Constraints:
        Affinity:
          exclude-any:
           blue
      Dynamic (active)
        Metric Type: DELAY, Path Accumulated Metric: 40
          FCCC:CCC1:AA22:: [Node-SID]
          FCCC:CCC1:AA33:: [Node-SID]
          FCCC:CCC1:AA11:: [Node-SID]
          FCCC:CCC1:AA11:E001:: [Adjacency-SID]
    Preference 1 (CLI):
```

```
      PM State: Unknown
      Dynamic (inactive)
        Inactive Reason: Perf Measure State Change to Pending
        Metric Type: TE, Path Accumulated Metric: 10
          FCCC:CCC1:C3:: [Node-SID]
  Attributes:
  Forwarding-ID: 16777217
  Per owner configs:
    CLI
      Binding SID: not configured
      Performance-measurement:
        liveness-detection
        invalidation-action down
  Stats:
    Packets: 0  Bytes: 0
  PM profile: Not configured
```

```
router# show segment-routing traffic-eng policy name PFP

Name: PFP (Color: 100 End-point: C02:1::1)
  Owners : CLI
  Status:
    Admin: up, Operational: up for 00:03:00 (since 04-17 10:46:06.552)
  Candidate-paths:
    Preference 1 (CLI):
      Per-flow Information (active):
        Forward        PDP     PDP  BSID RW
          Class      Color  Status   Status
        -------- ---------- ------- --------
               0          1      up  Pending
               1          2      up  Pending
      Default Forward Class: 0
  Attributes:
```

Example 2: Use the **show ip cef label-table <label> internal** command to view the PFP label details.

```
router# show ip cef label-table 16777218 internal
Label-FIB is Enabled
VRF Default
 3 prefixes (3/0 fwd/non-fwd)
 Table id 0x30000000
 Database epoch:        0 (3 entries at this epoch)

16777218 , epoch 0, refcnt 8, per-destination sharing
  sources: RR, Bnd-Lbl-SRv6-Pol
  subblocks:
    1 RR source [no flags]
    Binding Label SRv6 Policy: 16777218
      Policy-Name: PFP (16777218)  è PFP Policy Name
      Path: 0
        Flags: 00000000
        IPv6 Header Parameters
          TC: 0   Flow: 0       Hop Limit: 0
          Src: C01:1::1
          Dst: 16777217
        Segment List (0)
      Path: 1
        Flags: 00000000
        IPv6 Header Parameters
          TC: 0   Flow: 0       Hop Limit: 0
          Src: C01:1::1
          Dst: 16777217
```

```
    Segment List (0)
  Path: 2
    Flags: 00000000
    IPv6 Header Parameters
      TC: 0   Flow: 0       Hop Limit: 0
      Src: C01:1::1
      Dst: 16777217
    Segment List (0)
  Path: 3
    Flags: 00000000
    IPv6 Header Parameters
      TC: 0   Flow: 0       Hop Limit: 0
      Src: C01:1::1
      Dst: 16777217
    Segment List (0)
  Path: 4
    Flags: 00000000
    IPv6 Header Parameters
      TC: 0   Flow: 0       Hop Limit: 0
      Src: C01:1::1
      Dst: 16777217
    Segment List (0)
  Path: 5
    Flags: 00000000
    IPv6 Header Parameters
      TC: 0   Flow: 0       Hop Limit: 0
      Src: C01:1::1
      Dst: 16777217
    Segment List (0)
  Path: 6
    Flags: 00000000
    IPv6 Header Parameters
      TC: 0   Flow: 0       Hop Limit: 0
      Src: C01:1::1
      Dst: 16777217
    Segment List (0)
  Path: 7
    Flags: 00000000
    IPv6 Header Parameters
      TC: 0   Flow: 0       Hop Limit: 0
      Src: C01:1::1
      Dst: 16777217
    Segment List (0)
```

Example 3: Use the **show segment-routing traffic-eng cspf** command to view the CSPF details.

```
router# show segment-routing traffic-eng cspf ipv6 source A001::1 destination A006::1
metric-type delay

Path:
  HOP0: SRv6 NODE SID=F:1:6::
Path Cost = 10

CSPF result: Shortest Path Success (rc=8)
```

Example 4: Use the show prefix commands to display the color and binding SID associated with the BGP prefix path:

```
router# show bgp vpnv4 unicast vrf red 22.22.22.22
BGP routing table entry for 1:1:22.22.22.22/32, version 14
Paths: (1 available, best #1, table red)
Advertised to update-groups:
```

```
3
Refresh Epoch 1
3, imported path from 2:2:22.22.22.22/32 (global)
2023:1::1 (via default) from 1.1.1.3 (1.1.1.3)
Origin incomplete, metric 0, localpref 100, valid, internal, best
Extended Community: RT:1:1 RT:2:2 Color:10
Originator: 11.1.1.1, Cluster list: 1.1.1.3
binding SID: 16777217 (color - 10) (state - UP)
srv6 out-sid: FCCC:CCC1:AA88:E000::
rx pathid: 0, tx pathid: 0x0
Updated on Jun 12 2023 15:33:20 PST
```

```
router# show bgp vpnv6 unicast vrf red 2222::1/128
BGP routing table entry for [1:1]2222::1/128, version 13
Paths: (1 available, best #1, table red)
Advertised to update-groups:
3
Refresh Epoch 1
3, imported path from [2:2]2222::1/128 (global)
2023:1::1 (via default) from 1.1.1.3 (1.1.1.3)
Origin incomplete, metric 0, localpref 100, valid, internal, best
Extended Community: RT:2:2 Color:10
Originator: 11.1.1.1, Cluster list: 1.1.1.3
binding SID: 16777217 (color - 10) (state - UP)
srv6 out-sid: FCCC:CCC1:AA88:E001::
rx pathid: 0, tx pathid: 0x0
Updated on Jun 12 2023 15:33:20 PST
```

Example 5: Use the **show pce client peer** command to verify the PCEP connections:

```
router# show pce client peer
       PCC's peer database:
       --------------------
       Peer address: 192.168.0.6, Precedence: 255
       Client POLICY_MGR
        State up
        Capabilities: Stateful, Update, Segment-Routing, Instantiation, SRv6
```

Troubleshooting and Debugging SRv6-TE

Use the following commands to troubleshoot SRv6TE:

- **debug ip bgp sr-policy**
- **debug segment-routing traffic-eng**
  - **forwarding**: SR forwarding debug
  - **ha**: SR High-Availability debug
  - **path**: SR path debug
  - **pcalc**: SR pcalc debug
  - **policy**: SR policy debug
  - **topology**: SR topology debug

```
router# debug ip bgp sr-policy
*Apr 10 17:35:48.773: BGP(4): 2023:1::3 rcvd UPDATE w/ attr: nexthop 2023:1::1, origin ?,
localpref 100, metric 0, originator 11.1.1.1, clusterlist 1.1.1.3, merged path 3, AS_PATH
```

```
, extended community RT:1:1 RT:2:2 Color:10, PrefixSid attribute: SRV6 SID FCCC:CCC1:AA88::
*Apr 10 17:35:48.773: BGP(4): 2023:1::3 rcvd 2:2:22.22.22.22/32, label 2162163712 (0x80E00000)
*Apr 10 17:35:48.773: BGP SRv6 SID ATTR: blk 32 node 16 fun 16 arg 0 pos 16 off 48
*Apr 10 17:35:48.774: BGP-SR Policy (7F7911708510): Binding SID 10/2023:1::1/ request
*Apr 10 17:35:48.774: BGP(4): Revise route installing 1 of 1 routes for 22.22.22.22/32 ->
0.0.0.0(red) to red IP table
```

Performance Measurement for SRv6

Performance Measurement for SRv6

Feature Name	Release Information	Description
Performance Measurement for Segment Routing over IPv6	Cisco IOS XE Dublin 17.12.1a	This feature extends the performance measurement liveness to Segment Routing configuration over IPv6 data plane.

Performance Measurement Liveness for SRv6

From Cisco IOS XE 17.12.1a, Performance Measurement liveliness is extended to Segment Routing over IPv6 dataplane.

This feature enables Performance Measurement (PM) liveness detection for an SR policy on all the segment lists of every candidate path that are present in the forwarding table using PM probes. You can monitor the traffic path and efficiently detect any drop of traffic due to cable or hardware or configuration failures.

Prerequisites

- SRV6 must be enabled on all nodes before configuring PM for SRv6.

Restrictions

- By default, probes are sent every 3 seconds. You can increase the interval using the burst interval parameter (*). Reducing the probe interval to below 3 seconds is not recommended.

Configuring PM Liveness for SRv6

Use the following examples to configure PM liveliness for SRv6.

- Use the **liveness-detection** configuration under an SRV6 Policy to continuously monitor the state of SRV6 paths. This option provides only monitoring; no action is taken by the Policy Manager.
- Use the **invalidation-action down** configuration to configure the Policy Manager to:
  - Have the path programmed in HW only after it was validated with PM probes.
  - Continuously monitor the path.
  - Reoptimize to a different CP if PM probes stop working,

    or

bring the policy down if no other path is available.

Configure SRv6-TE PM Liveness under Policy

```
policy SRV6PM
  performance-measurement
   delay-measurement
    liveness-detection
     invalidation-action down
```

Configure Default Delay Profile for Liveness

```
performance-measurement
 delay-profile sr-policy
  probe
   liveness-detection
    multiplier 3
```

The following sections describe the recommended configurations for scaling deployment.

Configure PM Punt Policer for all PE Nodes

```
platform punt-policer sr-twamp-probe 3000
platform punt-policer sr-twamp-probe 3000 high
performance-measurement
  max-pps 3000
```

Configure Interface Queue for WAN Interface on Headend and Endpoint Nodes

```
interface Tunnel121
  hold-queue 10000 in
```

OR

```
interface GigabitEthernet0/0/1
  hold-queue 10000 in
```

Note The WAN interface includes physical interfaces and GRE-TP tunnels.

Verifying Performance Measurement for SRv6

Use the following show commands to verify PM configuration for SRv6.

Example 1: **`show performance-measurement sr-policy name <name>`**

```
device# show performance-measurement sr-policy name SRV6PM
SR Policy name: SRV6PM
  Color                          : 1
  Endpoint                       : C02:1::1
  Source                         : C01:1::1
  Profile name                   : Not configured
  Policy Update Timestamp        : 04-11 12:12:51.658
  Number of candidate-paths      : 2
  Candidate-Path:
    Preference                   : 1
    Protocol-origin              : CLI
    Discriminator                : 0
```

```
    Number of segment-lists       : 1
    Number of atomic paths        : 1
    Number of live UP atomic paths: 0
    Number of live Unknown atomic : 0
    Max Pkts per Burst            : 1500
    Max Pkts per Probe            : 15000
    AP Min Run per Probe          : 3
    Round-robin bursts            : 1
    Round-robin probes            : 1
    Last advertisement:
      Advertised at: 12:12:06 04-11 2023 (516007 seconds ago)
Atomic path:
        Hops                      : C2:1::1, C3:1::1, C1:1::1
                                  : 2021:2::1
        Labels                    : FCCC:CCC1:AA22:AA33:AA11:E004::
        Outgoing Interface        : Ethernet0/2
        Max IP MTU                : 1500
        Next Hop                  : FE80::A8BB:CCFF:FE00:FA10
        Destination               : C02:1::1
        Session ID                : 8
        Last advertisement:
          No advertisements have occured
        Next advertisement:
          Aggregated delays (uSec): avg: 2744, min: 1480, max: 21676, variance: 1172
          Rolling average (uSec): 2744
        Last probe:
          Packets Sent: 10, received: 10
          Measured delays (uSec): avg: 1666, min: 1480, max: 1853, variance: 186
        Current probe:
          Packets Sent: 2, received: 2
          Measured delays (uSec): avg: 6192, min: 1619, max: 10765, variance: 4573
        Probe samples:
          Packet Rx Timestamp       Measured Delay (nsec)
          11:37:29 04-17 2023       1619000
          11:37:26 04-17 2023       10765000
```

Example 2: **`show performance-measurement sr-policy name <name> d p v | s Liveness`**

```
device# show performance-measurement sr-policy name SRV6PM d p v | s Liveness
        Liveness Detection:
          Session Creation Timestamp: 04-11 12:10:49.981
          Session State: Down
          Last State Change Timestamp: 04-11 12:12:51.656
          Missed count   [consecutive]: 84752
          Received count [consecutive]: 0
          Backoff                     : 1
          Unique Path Name            : Path-10
          Loss in Last Interval       : 100 % [TX: 7 RX: 0]
        Liveness Detection:
          Session Creation Timestamp: 04-11 12:12:36.636
          Session State: Up
          Last State Change Timestamp: 04-11 12:12:36.728
          Missed count   [consecutive]: 0
          Received count [consecutive]: 84717
          Backoff                     : 0
          Unique Path Name            : Path-12
          Loss in Last Interval       : 0 % [TX: 7 RX: 7]
        Liveness Detection:
          Session Creation Timestamp: 04-11 12:12:36.636
          Session State: Up
          Last State Change Timestamp: 04-11 12:12:36.728
          Missed count   [consecutive]: 0
          Received count [consecutive]: 84717
```

```
          Backoff                     : 0
          Unique Path Name            : Path-13
          Loss in Last Interval       : 0 % [TX: 7 RX: 7]
```

Example 3: **show segment-routing traffic-eng policy all type per-destination**

```
device# show segment-routing traffic-eng policy all type per-destination
Name: SRV6PM (Color: 1 End-point: C02:1::1)
  Owners : CLI
  Status:
    Admin: up, Operational: up for 70:55:04 (since 04-11 12:10:05.054)
  Candidate-paths:
    Preference 2 (CLI):
      PM State: Up
      Constraints:
        Affinity:
          exclude-any:
           blue
      Dynamic (active)
        Metric Type: DELAY, Path Accumulated Metric: 40
          FCCC:CCC1:AA22:: [Node-SID]
          FCCC:CCC1:AA33:: [Node-SID]
          FCCC:CCC1:AA11:: [Node-SID]
          FCCC:CCC1:AA11:E001:: [Adjacency-SID]
    Preference 1 (CLI):
      PM State: Unknown
      Dynamic (inactive)
        Inactive Reason: Perf Measure State Change to Pending
        Metric Type: TE, Path Accumulated Metric: 10
          FCCC:CCC1:C3:: [Node-SID]
  Attributes:
```

Example 4: **show performance-measurement history interfaces adv**

```
device# show performance-measurement history interfaces adv
Interface Name: Ethernet0/0 (ifh: 0x2)
  Delay-Measurement history (uSec):
    Session ID: 1
    Advertisement Timestamp   Average   Min       Max       Action
    12:10:05 04-11 2023       10        10        10        CFG
Interface Name: Ethernet0/1 (ifh: 0x3)
  Delay-Measurement history (uSec):
    Session ID: 2
    Advertisement Timestamp   Average   Min       Max       Action
    12:10:05 04-11 2023       15        15        15        CFG
Interface Name: Tunnel100 (ifh: 0x15)
  Delay-Measurement history (uSec):
    Session ID: 3
    Advertisement Timestamp   Average   Min       Max       Action
    13:10:55 04-13 2023       603       307       969       PER-MIN
    13:04:46 04-13 2023       8696      1384      18908     PER-MIN
    10:31:05 04-13 2023       6897      377       38335     PER-MIN
    10:26:56 04-13 2023       6792      1802      13778     PER-MIN
    12:12:26 04-11 2023       3018      363       14081     FIRST
Interface Name: Tunnel101 (ifh: 0x16)
  Delay-Measurement history (uSec):
    Session ID: 4
    Advertisement Timestamp   Average   Min       Max       Action
    12:12:16 04-11 2023       1841      263       8400      FIRST
```

Example 5: **show performance-measurement history sr-policy liveness-notification**

```
device# show performance-measurement history sr-policy liveness-notification
SR Policy name: pdp-voice
  Candidate-Path:
    Preference                : 10
    Protocol-origin           : CLI
    Discriminator             : 0
    Active                    : No
    Segment-list:
      Name                    : SL13
      Atomic path:
        Hops                  : A006::1
        Labels                : ::
        Outgoing Interface    : Tunnel16
        Next Hop              : 1634::6
        Destination           : A006::1
        Delay-Measurement:
          Session ID          : 16
          Liveness state change timestamp                   New State
          04:20:25 01-15 2023                               Up

  Candidate-Path:
    Preference                : 50
    Protocol-origin           : CLI
    Discriminator             : 0
    Active                    : No
    Segment-list:
      Name                    : SL12
      Atomic path:
        Hops                  : ::, ::, 5646::5
        Labels                : F:1:2:5:E003::
        Outgoing Interface    : Tunnel12
        Next Hop              : 1211::2
        Destination           : A006::1
        Delay-Measurement:
          Session ID          : 23
          Liveness state change timestamp                   New State
          04:30:19 01-15 2023                               Up

  Candidate-Path:
    Preference                : 100
    Protocol-origin           : CLI
    Discriminator             : 0
    Active                    : Yes
    Segment-list:
      Name                    : SL11
      Atomic path:
        Hops                  : ::, ::, 5631::5
        Labels                : F:1:4:5:E002::
        Outgoing Interface    : GigabitEthernet2
        Next Hop              : FE80::5054:FF:FE1A:DD62
        Destination           : A006::1
        Delay-Measurement:
          Session ID          : 14
          Liveness state change timestamp                   New State
          04:20:17 01-15 2023                               Up
```

Example 6: `show isis teapp`

```
device# show isis teapp
Tag null:
   ISIS TEAPP Information: Topology(ID:0x0) Type:SRTE, Enabled:1, Router ID:0.0.0.0 IPv6
Router ID:C01:1::1
 Topology Id:0x0 Teapp_type:SRTE
```

```
   Interface(hdl:0x2): Ethernet0/0
    Affinity: set 1, affinity_bits 8
    TE Metric: set 1, te_metric 1000
    Extended Affinity: set 1, length 1, ext_affinity_bits: 8
 Topology Id:0x0 Teapp_type:SRTE
   Interface(hdl:0x3): Ethernet0/1
    Affinity: set 1, affinity_bits 8
    TE Metric: set 1, te_metric 1000
    Extended Affinity: set 1, length 1, ext_affinity_bits: 8
   ISIS TE Attr PM Information:
    Et0/0: IDB num:2 Min:10 Max:10 Min-max-anomaly:0 Avg:10 Avg-anomaly:0 Var:0
          Is-Loss-set:0 Loss:0 Loss-anomaly:0
    Et0/1: IDB num:3 Min:15 Max:15 Min-max-anomaly:0 Avg:15 Avg-anomaly:0 Var:0
          Is-Loss-set:0 Loss:0 Loss-anomaly:0
    Tu100: IDB num:21 Min:307 Max:969 Min-max-anomaly:0 Avg:603 Avg-anomaly:0 Var:109
          Is-Loss-set:0 Loss:0 Loss-anomaly:0
    Tu101: IDB num:22 Min:263 Max:8400 Min-max-anomaly:0 Avg:1841 Avg-anomaly:0 Var:1042
          Is-Loss-set:0 Loss:0 Loss-anomaly:0
device#
```

Example 7: `show performance-measurement responder summary`

```
device# show performance-measurement responder summary
  Total interfaces                              : 5
  Total query packets received                  : 509200
  Total reply packets sent                      : 509200
  Total reply packets sent errors               : 0
  Total URO TLV not present errors              : 0
  Total invalid port number errors              : 0
  Total no source address errors                : 0
  Total no return path errors                   : 0
  Total unsupported querier control code errors : 0
  Total unsupported timestamp format errors     : 0
  Total timestamp not available errors          : 0
  Total unsupported mandatory TLV errors        : 0
  Total invalid packet errors                   : 0
  Total loss probe color errors                 : 0
  Current rate                                  : 1 pkts/sec
  Rate high water mark                          : 3 pkts/sec
```

Example 8: `show monitor event-trace perf_measure all`

```
device# show monitor event-trace perf_measure all
Perf Measure error events:
Perf Measure event events:
*Apr 11 17:10:05.115: PM-TRACE-IGP-ADV :flood Ethernet0/0 10 10 10
*Apr 11 17:10:05.116: PM-TRACE-IGP-ADV :flood Ethernet0/1 15 15 15
*Apr 11 17:12:16.492: PM-TRACE-IGP-ADV :flood Tunnel101 263 8400 1841
*Apr 11 17:12:26.582: PM-TRACE-IGP-ADV :flood Tunnel100 363 14081 3018
*Apr 13 15:26:56.861: PM-TRACE-IGP-ADV :flood Tunnel100 1802 13778 6792
*Apr 13 15:31:05.510: PM-TRACE-IGP-ADV :flood Tunnel100 377 38335 6897
*Apr 13 18:04:46.608: PM-TRACE-IGP-ADV :flood Tunnel100 1384 18908 8696
*Apr 13 18:10:55.245: PM-TRACE-IGP-ADV :flood Tunnel100 307 969 603
interrupt context allocation count = 0
```

Example 9: `show performance-measurement summary`

```
device# show performance-measurement summary
Total interfaces                            : 4
Total SR Policies                           : 2
Total endpoints                             : 0
Maximum PPS                                 : 2000 pkts/sec
```

```
Dual-color gre bit-position                  : 9 - Failed, last success 0
Interface Delay-Measurement:
  Total sessions                             : 4
  Counters:
    Packets:
      Total sent                             : 338865
      Total received                         : 338861
    Errors:
      Total sent errors                      : 14
      Total received errors                  : 0
    Probes:
      Total started                          : 33892
      Total completed                        : 33884
      Total incomplete                       : 4
      Total advertisements                   : 8
SR Policy Delay-Measurement:
  Total sessions                             : 4
  Counters:
    Packets:
      Total sent                             : 339076
      Total received                         : 169602
    Errors:
      Total sent errors                      : 0
      Total received errors                  : 0
    Probes:
      Total started                          : 33912
      Total completed                        : 16964
      Total incomplete                       : 16948
      Total advertisements                   : 243
```

SRv6 OAM

SRv6 Operations, Administration, and Maintenance

From Cisco IOS XE 17.12.1a, Operations, Administration, and Maintenance (OAM) functionality is suported by SRv6, using Segment Lists and SRv6 Policy.

Restrictions for SRv6

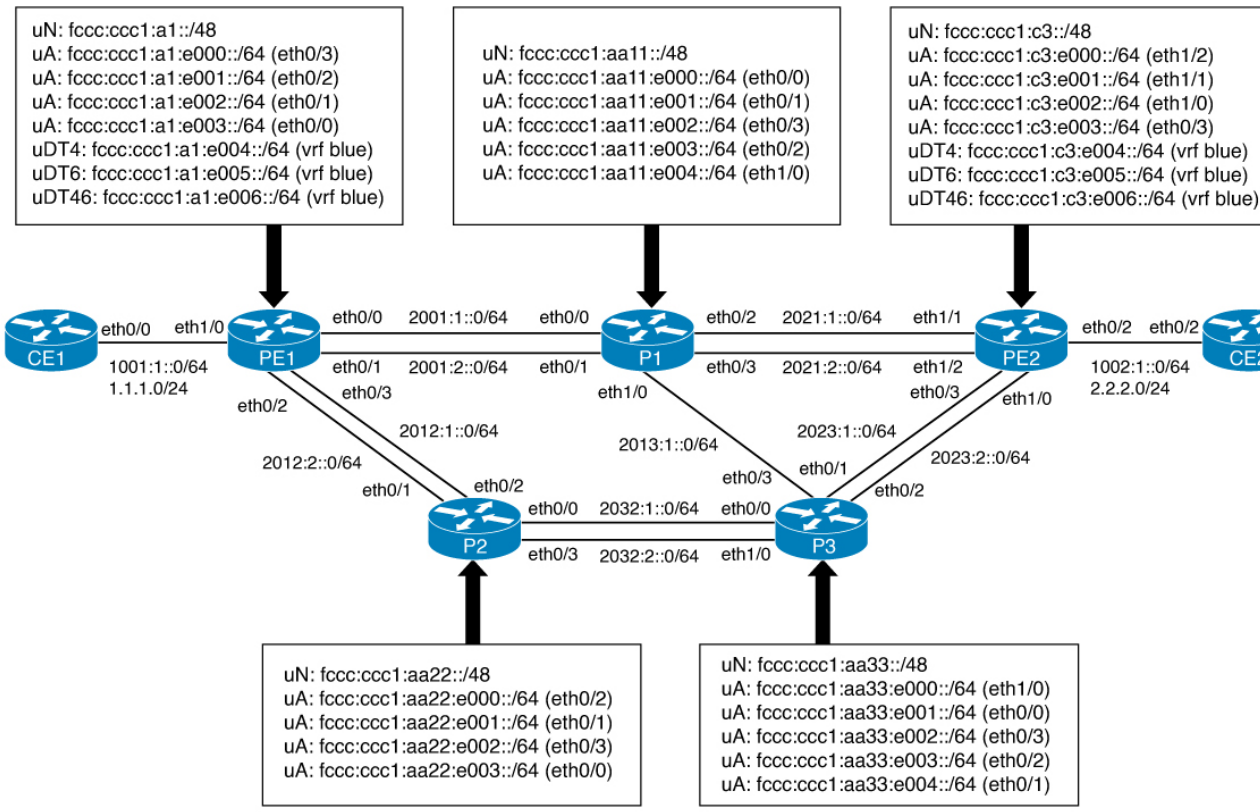
- Traceroute to IPv4 VRF does not display the core PE nodes.
- Ping or Traceroute IPv4 VRF using custom SID-List is not supported.

Information About SRv6 OAM

Operations, Administration, and Maintenance (OAM) helps service providers to monitor SRv6 paths and quickly isolate forwarding problems to assist with fault detection and troubleshooting in the network.

The following figure provides a sample topology for SRv6 OAM.

Figure 5: Sample SRv6 OAM Topology



Operating SRv6 OAM

SRv6 OAM involves the following operations:

- IPv6 Ping/Traceroute CE-CE across SRv6 Core
- IPv4 Ping/Traceroute CE-CE across SRv6 Core
- IPv6 Ping/Traceroute PE-CE across SRv6 Core
- IPv4 Ping/Traceroute PE-CE across SRv6 Core
- IPv6 SID Ping/Traceroute
- IPv6 VRF Ping/Traceroute using custom SIDs

The following examples reference the topology in Figure x.

Operate IPv6 Ping/Traceroute CE-CE across SRv6 Core

Use the following example to operate Ping/Traceroute CE-CE connected IPv6 interface IP:

```
CE1#ping 1002:1::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1002:1::2, timeout is 2 seconds:
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/7 ms
CE1#

CE1#traceroute 1002:1::2 probe 1
Type escape sequence to abort.
Tracing the route to 1002:1::2
  1 1001:1::1 1 msec
  2 1002:1::2 1 msec
CE1#
```

Operate IPv4 Ping/Traceroute CE-CE across SRv6 Core

Use the following example to operate Ping/Traceroute CE-CE connected IPv4 interface IP:

```
CE1#ping 2.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
CE1#

CE1#traceroute 2.2.2.2 probe 1
Type escape sequence to abort.
Tracing the route to 2.2.2.2
VRF info: (vrf in name/id, vrf out name/id)
  1 1.1.1.1 1 msec
  2 2.2.2.2 1 msec
CE1#
```

Operate IPv6 Ping/Traceroute PE-CE across SRv6 Core

Use the following example to operate Ping/Traceroute CE's IPv6 interface from PE's VRF interface via SRv6 core:

```
PE1#ping vrf blue 1002:1::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1002:1::2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
PE1#

PE1#traceroute vrf blue 1002:1::2 probe 1
Type escape sequence to abort.
Tracing the route to 1002:1::2
  1 2001:1::2 1 msec
  2 2021:2::2 1 msec
  3 1002:1::2 1 msec
PE1#
```

Operate IPv4 Ping/Traceroute PE-CE across SRv6 Core

Use the following example to operate Ping/Traceroute CE's IPv4 interface from PE's VRF interface via SRv6 core:

```
PE1#ping vrf blue 2.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
PE1#

PE1#traceroute vrf blue 2.2.2.2 probe 1
Type escape sequence to abort.
```

```
Tracing the route to 2.2.2.2
VRF info: (vrf in name/id, vrf out name/id)
  1  *
  2  *
  3 2.2.2.2 1 msec
PE1#
```

Note The IPv4 ping displays "*" instead of IPv6 hops.

Operate IPv6 SID Ping/Traceroute

Use the following example to operate Ping/Traceroute PE2's node SID SRv6 SID from PE1:

```
PE1#ping FCCC:CCC1:C3::
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FCCC:CCC1:C3::, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
PE1#

PE1#traceroute FCCC:CCC1:C3:: probe 1
Type escape sequence to abort.
Tracing the route to FCCC:CCC1:C3::
  1 2001:1::2 0 msec
  2 2021:2::2 0 msec
PE1#
```

Operate IPv6 VRF Ping/Traceroute using Custom SIDs

Use the following example to operate Ping/Traceroute CE2 IPv6 interface from PE1 using custom SRv6 SID list:

The SID in this traceroute goes from PE1 to PE2 via P1, P2 and P3, and finally to CE2.

- The first SID is from PE1 to P1.
- The next SID, from P1 to P2, is an ECMP path via P3 (P1 -> P3 -> P2).
- The next SID, to reach the VPN-SID PE2, is P2 -> P3 -> PE2.
- The last SID is to reach CE2 from PE2 (PE2 -> CE2).

```
PE1#ping srv6 vrf blue 1002:1::2 via segment-list FCCC:CCC1:AA11:AA22:C3:E005::
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1002:1::2 via [
 FCCC:CCC1:AA11:AA22:C3:E005::], timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
PE1#

PE1# traceroute srv6 vrf blue 1002:1::2 via segment-list FCCC:CCC1:AA11:AA22:C3:E005:: probe
 1
Type escape sequence to abort.
Tracing the route to 1002:1::2 via [
 FCCC:CCC1:AA11:AA22:C3:E005::]
  1 2001:1::2 1 msec
  2 2013:1::2 1 msec
  3 2032:2::1 1 msec
  4 2032:1::2 1 msec
```

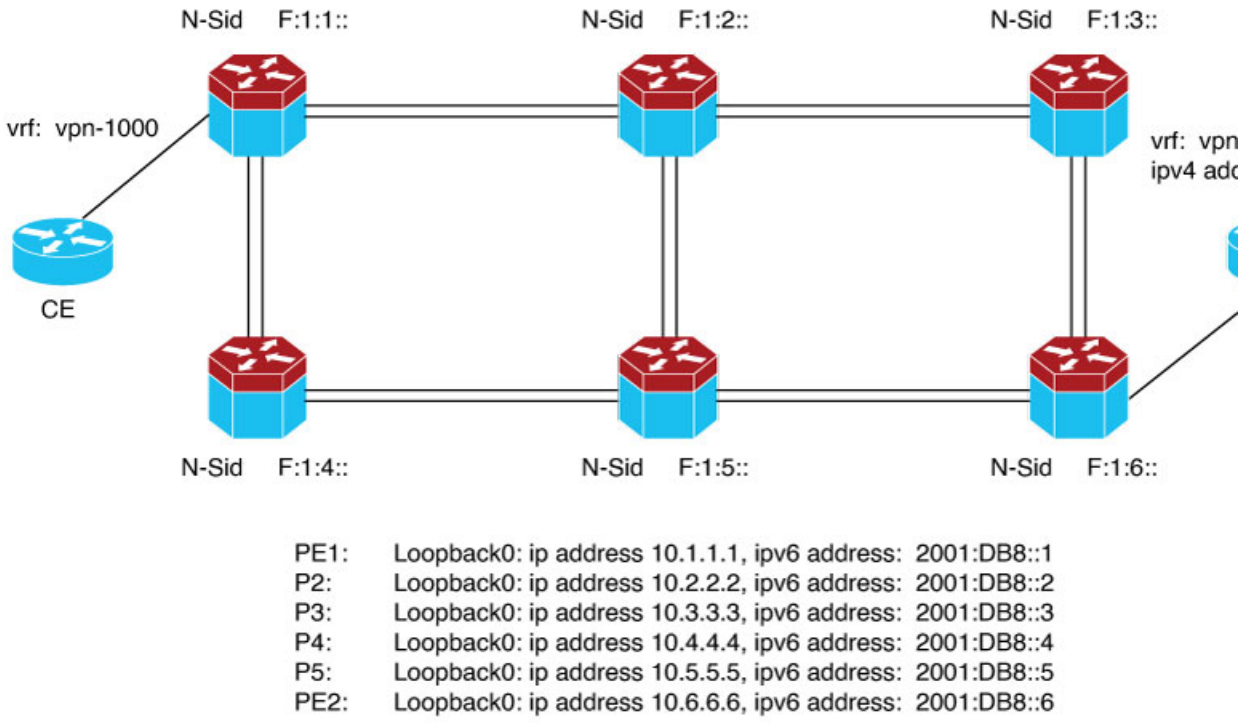
```
  5 2023:1::1 1 msec
  6 1002:1::2 1 msec
PE1#
```

From Cisco IOS XE 17.16.1a, the following operational commands are available:

- `ping srv6 <ipv4-destination> via segment-list […]`
- `traceroute srv6 <ipv4-destination> via segment-list […]`
- `ping srv6 vrf <vrf-name> <ipv4-destination> via segment-list […]`
- `traceroute srv6 vrf <vrf-name> <ipv4-destination> via segment-list […]`
- `traceroute vrf <vrf-name> <ipv4-destination> […]`

The examples below are based on the topology in ***Fig. SRv6 OAM Topology***.

Figure 6: SRv6 OAM Topology




```
PE1#ping srv6 10.6.6.6 via segment-list F:1:4::,F:1:5::,F:1:2::,F:1:3::,F:1:6::
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.6.6.6 via [
 F:1:4::
 F:1:5::
 F:1:2::
 F:1:3::
 F:1:6::], timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
```

```
PE1#traceroute srv6 10.6.6.6 via segment-list F:1:4::,F:1:5::,F:1:2::,F:1:3::,F:1:6:: probe
 1
Type escape sequence to abort.
```

```
Tracing the route to 10.6.6.6 via [
 F:1:4::
 F:1:5::
 F:1:2::
 F:1:3::
 F:1:6::]
VRF info: (vrf in name/id, vrf out name/id)
  1 1411::4 1 msec
  2 4502::5 1 msec
  3 2511::2 1 msec
  4 2311::3 1 msec
  5 3612::6 1 msec
  6 36.11.0.6 1 msec
```

```
PE1#ping srv6 vrf vpn-1000 10.68.0.6 via segment-list
F:1:4::,F:1:5::,F:1:2::,F:1:3::,F:1:6:E002::
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.68.0.6 via [
 F:1:4::
 F:1:5::
 F:1:2::
 F:1:3::
 F:1:6:E002::], timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

```
PE1#traceroute srv6 vrf vpn-1000 10.68.0.6 via segment-list
F:1:4::,F:1:5::,F:1:2::,F:1:3::,F:1:6:E002:: probe 1
Type escape sequence to abort.
Tracing the route to 10.68.0.6 via [
 F:1:4::
 F:1:5::
 F:1:2::
 F:1:3::
 F:1:6:E002::]
VRF info: (vrf in name/id, vrf out name/id)
  1 1411::4 0 msec
  2 4502::5 1 msec
  3 2511::2 1 msec
  4 2311::3 1 msec
  5 3612::6 1 msec
  6 68.0.0.6 1 msec
```

```
PE1#traceroute vrf vpn-1000 10.68.0.6 probe 1
Type escape sequence to abort.
Tracing the route to 10.68.0.6
VRF info: (vrf in name/id, vrf out name/id)
  1 1411::4 1 msec
  2 4501::5 1 msec
  3 5611::6 0 msec
  4 10.68.0.6 2 msec
```

Support for SRv6 OAM-TE

From Cisco IOS XE 17.15.1a, the following operational commands provide support for SRv6 OAM traffic engineering:

```
# ping srv6 policy <[<policy-name>] | [color <color-value> endpoint <end-point>]> […]
```

```
# traceroute srv6 policy <[<policy-name>] | [color <color-value> endpoint <end-point>]> […]
```

Note These commands do not support SID list fragmentation.

Use the following examples to operate Ping/Traceroute for OAM traffic engineering:

```
device#ping srv6 policy SRV6PM
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to C02:1::1 via [
 FCCC:CCC1:AA22:AA33:AA11:E001::], timeout is 2 seconds:
Packet sent with a source address of C01:1::1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/9 ms
device#
```

```
device#ping srv6 policy color 1 endpoint C02:1::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to C02:1::1 via [FCCC:CCC1:AA22:AA33:AA11:E001::],
timeout is 2 seconds:
Packet sent with a source address of C01:1::1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
device#
```

```
device#traceroute srv6 policy SRv6PM
Type escape sequence to abort.
Tracing the route to C02:1::1 via [FCCC:CCC1:AA22:AA33:AA11:E001::]
  1 2012:1::2 1 msec
    2012:2::2 1 msec
    2012:1::2 1 msec
  2 2032:2::2 3 msec 1 msec 3 msec
  3 2013:1::1 0 msec 0 msec 0 msec
  4 2021:2::2 1 msec 1 msec 1 msec
device#
```

```
device#traceroute srv6 policy color 1 endpoint C02:1::1
Type escape sequence to abort.
Tracing the route to C02:1::1 via [FCCC:CCC1:AA22:AA33:AA11:E001::]
  1 2012:1::2 1 msec
    2012:2::2 1 msec
    2012:1::2 0 msec
  2 2032:2::2 1 msec 1 msec 0 msec
  3 2013:1::1 1 msec 1 msec 0 msec
  4 2021:2::2 1 msec 1 msec 0 msec
device#
```

Troubleshooting and Debugging SRv6 OAM-TE

Use the following commands to debug SRv6 configuration:

- `# debug platform hardware qfp active feature cef-mpls datapath ipv6 all`
- `# debug srv6 all`
- `# debug isis srv6`
- `# debug bgp ipv6 updates`
- `# debug bgp ipv6 addpath`
- `# debug ip bgp srv6`

- `# debug isis fast-reroute path-selection | ti-lfa | trigger`
- `# debug isis ipv6 microloop`
- `# debug ipv6 packet`