Segment Routing Configuration Guide, Cisco IOS XE Release 3S

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

CHAPTER 1

Introduction to Segment Routing

Segment Routing (SR) is a flexible, scalable way of doing source routing.

Overview of Segment Routing

Segment Routing (SR) is a flexible, scalable way of doing source routing. The source chooses a path and encodes it in the packet header as an ordered list of segments. Segments are identifier for any type of instruction. Each segment is identified by the segment ID (SID) consisting of a flat unsigned 32-bit integer. Segment instruction can be:

- Go to node N using the shortest path
- Go to node N over the shortest path to node M and then follow links Layer 1, Layer 2, and Layer 3
- Apply service S

With segment routing, the network no longer needs to maintain a per-application and per-flow state. Instead, it obeys the forwarding instructions provided in the packet.

Segment Routing relies on a small number of extensions to Cisco Intermediate System-to-Intermediate System (IS-IS) and Open Shortest Path First (OSPF) protocols. It can operate with an MPLS (Multiprotocol Label Switching) or an IPv6 data plane, and it integrates with the rich multi service capabilities of MPLS, including Layer 3 VPN (L3VPN), Virtual Private Wire Service (VPWS), Virtual Private LAN Service (VPLS), and Ethernet VPN (EVPN).

Segment routing can be directly applied to the Multiprotocol Label Switching (MPLS) architecture with no change in the forwarding plane. Segment routing utilizes the network bandwidth more effectively than traditional MPLS networks and offers lower latency. A segment is encoded as an MPLS label. An ordered list of segments is encoded as a stack of labels. The segment to process is on the top of the stack. The related label is popped from the stack, after the completion of a segment.

Segment routing can be applied to the IPv6 architecture with a new type of routing extension header. A segment is encoded as an IPv6 address. An ordered list of segments is encoded as an ordered list of IPv6 addresses in the routing extension header. The segment to process is indicated by a pointer in the routing extension header. The pointer is incremented, after the completion of a segment.

Segment Routing provides automatic traffic protection without any topological restrictions. The network protects traffic against link and node failures without requiring additional signaling in the network. Existing IP fast re-route (FRR) technology, in combination with the explicit routing capabilities in Segment Routing guarantees full protection coverage with optimum backup paths. Traffic protection does not impose any additional signaling requirements.

How Segment Routing Works

A router in a Segment Routing network is capable of selecting any path to forward traffic, whether it is explicit or Interior Gateway Protocol (IGP) shortest path. Segments represent subpaths that a router can combine to form a complete route to a network destination. Each segment has an identifier (Segment Identifier) that is distributed throughout the network using new IGP extensions. The extensions are equally applicable to IPv4 and IPv6 control planes. Unlike the case for traditional MPLS networks, routers in a Segment Router network do not require Label Distribution Protocol (LDP) and Resource Reservation Protocol - Traffic Engineering (RSVP-TE) to allocate or signal their segment identifiers and program their forwarding information.

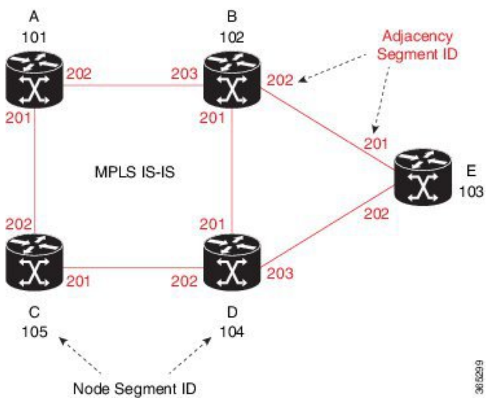
Each router (node) and each link (adjacency) has an associated segment identifier (SID). Node segment identifiers are globally unique and represent the shortest path to a router as determined by the IGP. The network administrator allocates a node ID to each router from a reserved block. On the other hand, an adjacency segment ID is locally significant and represents a specific adjacency, such as egress interface, to a neighboring router. Routers automatically generate adjacency identifiers outside of the reserved block of node IDs. In an MPLS network, a segment identifier is encoded as an MPLS label stack entry. Segment IDs direct the data along a specified path. There are two kinds of segment IDS:

- **Prefix SID—** A segment ID that contains an IP address prefix calculated by an IGP in the service provider core network. Prefix SIDs are globally unique. A node SID is a special form of prefix SID that contains the loopback address of the node as the prefix. It is advertised as an index into the node specific SR Global Block or SRGB.
- **Adjacency SID—** A segment ID that contains an advertising router’s adjacency to a neighbor. An adjacency SID is a link between two routers. Since the adjacency SID is relative to a specific router, it is locally unique.

Examples for Segment Routing

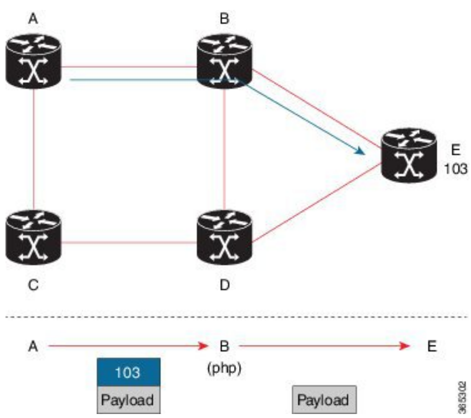
The following figure illustrates an MPLS network with five routers using Segment Routing, IS-IS, a label range of 100 to 199 for node IDs, and 200 and higher for adjacency IDs. IS-IS would distribute IP prefix reachability alongside segment ID (the MPLS label) across the network.

Figure 1: An MPLS Network with Five Routers Using Segment Routing



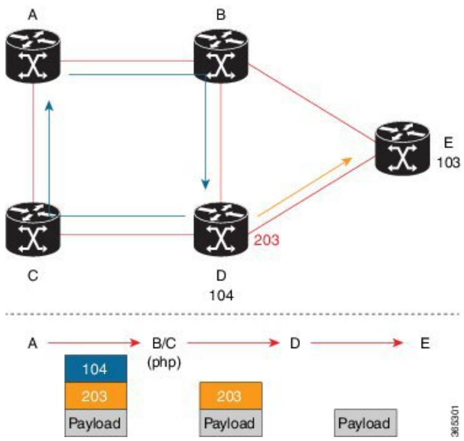
In the previous example, any router sending traffic to router E would push label 103 (router E node segment identifier) to forward traffic using the IS-IS shortest path. The MPLS label-swapping operation at each hop preserves label 103 until the packet arrives at E (Figure 2). On the other hand, adjacency segments behave differently. For example, if a packet arrives at Router D with a top-of-stack MPLS label of 203 (D-to-E adjacency segment identifier), Router D would pop the label and forward the traffic to Router E.

Figure 2: MPLS Label-Swapping Operation



Segment identifiers can be combined as an ordered list to perform traffic engineering. A segment list can contain several adjacency segments, several node segments, or a combination of both depending on the forwarding requirements. In the previous example, Router A could alternatively push label stack (104, 203) to reach Router E using the shortest path and all applicable ECMPs to Router D and then through an explicit interface onto the destination (Figure 3). Router A does not need to signal the new path, and the state information remains constant in the network. Router A ultimately enforces a forwarding policy that determines which flows destined to router E are switched through a particular path.

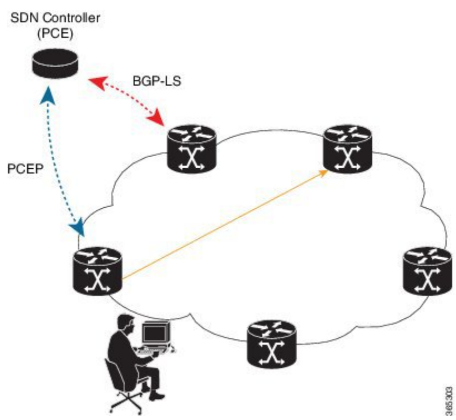
Figure 3: Router E Destination Path



Benefits of Segment Routing

- **Ready for SDN—** Segment Routing is a compelling architecture conceived to embrace Software-Defined Network (SDN) and is the foundation for Application Engineered Routing (AER). It strikes a balance between network-based distributed intelligence, such as automatic link and node protection, and controller-based centralized intelligence, such as traffic optimization. It can provide strict network performance guarantees, efficient use of network resources, and very high scalability for application-based transactions. The network uses minimal state information to meet these requirements. Segment routing can be easily integrated with a controller-based SDN architecture. Below figure illustrates a sample SDN scenario where the controller performs centralized optimization, including bandwidth admission control. In this scenario, the controller has a complete picture of the network topology and flows. A router can request a path to a destination with certain characteristics, for example, delay, bandwidth, diversity. The controller computes an optimal path and returns the corresponding segment list, such as an MPLS label stack, to the requesting router. At that point, the router can inject traffic with the segment list without any additional signaling in the network.

Figure 4: SDN Controller



- In addition, segment lists allow complete network virtualization without adding any application state to the network. The state is encoded in the packet as a list of segments. Because the network only maintains segment state, it can support a large number - and a higher frequency - of transaction-based application requests without creating any burden on the network.
- **Simplified—**
    - When applied to the MPLS data plane, Segment Routing offers the ability to tunnel MPLS services (VPN, VPLS, and VPWS) from an ingress provider edge to an egress provider edge without any other protocol than an IGP (ISIS or OSPF).
    - Simpler operation without separate protocols for label distribution (for example, no LDP or RSVP).
    - No complex LDP or IGP synchronization to troubleshoot.
    - Better utilization of installed infrastructure, for lower capital expenditures (CapEx), with ECMP-aware shortest path forwarding (using node segment IDs).
- **Supports Fast Reroute (FRR)—** Deliver automated FRR for any topology. In case of link or node failures in a network, MPLS uses the FRR mechanism for convergence. With segment routing, the convergence time is sub-50-msec.
- **Large-scale Data Center**-
    - Segment Routing simplifies MPLS-enabled data center designs using Border Gateway Protocol (BGP) RFC 3107 - IPv4 labeled unicast among Top-of-the-Rack/Leaf/Spine switches.
    - BGP distributes the node segment ID, equivalent to IGP node SID.
    - Any node within the topology allocates the same BGP segment for the same switch.
    - The same benefits are provided as for IGP node SID: ECMP and automated FRR (BGP PIC(Prefix Independent Convergence).
    - This is a building block for traffic engineering - SR TE data center fabric optimization.
- **Scalable—**

    - Avoid thousands of labels in LDP database.
    - Avoid thousands of MPLS Traffic Engineering LSP's in the network.
    - Avoid thousands of tunnels to configure.

- **Dual-plane Networks—**
    - Segment Routing provides a simple solution for disjointness enforcement within a so-called “dual-plane” network, where the route to an edge destination from a given plane stays within the plane unless the plane is partitioned.
    - An additional SID “anycast” segment ID allows the expression of macro policies such as: "Flow 1 injected in node A toward node Z must go via plane 1" and "Flow 2 injected in node A towards node Z must go via plane 2."

- **Centralized Traffic Engineering—**
    - Controllers and orchestration platforms can interact with Segment Routing traffic engineering for centralized optimization, such as WAN optimization.
    - Network changes such as congestion can trigger an application to optimize (recompute) the placement of segment routing traffic engineering tunnels.
    - Segment Routing tunnels are dynamically programmed onto the network from an orchestrator using southbound protocols like PCE.
    - Agile network programming is possible since Segment Routing tunnels do not require signaling and per-flow state at midpoints and tail end routers.

- **Egress Peering Traffic Engineering (EPE)—**
    - Segment Routing allows centralized EPE.
    - A controller instructs an ingress provider edge and content source to use a specific egress provider edge and specific external interface to reach a destination.
    - BGP “peering” segment IDs are used to express source-routed inter-domain paths.
    - Controllers learn BGP peering SIDs and the external topology of the egress border router through BGP Link Status (BGP-LS) EPE routes.
    - Controllers program ingress points with a desired path.

- **Plug-and-Play deployment—** Segment routing tunnels are interoperable with existing MPLS control and data planes and can be implemented in an existing deployment.

Segment Routing Global Block

Segment Routing Global Block (SRGB) is the range of labels reserved for segment routing. SRGB is local property of an segment routing node. In MPLS, architecture, SRGB is the set of local labels reserved for global segments. In segment routing, each node can be configured with a different SRGB value and hence the absolute SID value associated to an IGP Prefix Segment can change from node to node.

The SRGB default value is 16000 to 23999. The SRGB can be configured as follows:

```
Device(config)# router isis 1
Device(config-isis)#segment-routing global-block 45000 55000
```

The SRGB label value is calculated as follows:

- If the platform supports 1000000 labels or more, the SRGB value is from 900000 to 900000 + 2^16.
- If the platform supports less than 1000000 labels, the SRGB value is the last 2^16 labels.

Restrictions:

- The SRGB size cannot be more than 2^16.
- The SRGB upper bound cannot exceed the platform capability.
- The SRGB cannot be configured to be the same value as the default SRGB. So SRGB cannot be configured for 16000 to 23999.

Segment Routing Global Block

This chapter explains the concept of creating a block of labels reserved for a router using segment routing. This block of reserved labels is known as the Segment Routing Global Block (SRGB).

Adjacency Segment Identifiers

The Adjacency Segment Identifier (adj-SID) is a local label that points to a specific interface and a next hop out of that interface. No specific configuration is required to enable adj-SIDs. Once segment routing is enabled over IS-IS for an address-family, for any interface that IS-IS runs over, the address-family automatically allocates an adj-SID towards every neighbor out of that interface.

Note Only IPV4 address-family supports allocating adj-SIDs.

Prefix Segment Identifiers

A prefix segment identifier (SID) identifies a segment routing tunnel leading to the destination represented by a prefix. The maximum prefix SID value is 2^16 - 1.

A prefix SID is allocated from the Segment Routing Global Block (SRGB). The prefix SID value translates to a local MPLS label, whose value is calculated as below:

- If the platform supports 1000000 labels or more, then the MPLS label corresponding to the prefix SID value is 900000 + *sid-value*.
- If the platform supports less than 1000000 labels, then the MPLS label corresponding to the prefix SID value is *maximum-supported-label-value* - 2^16 + *sid-value*.

When a prefix SID value *x* is configured, the prefix SID translates to a label value equivalent to *x* + lower boundary of SRGB. For example, in the platform supporting 1000000 MPLS labels or more if the default

SRGB is used, configuring a prefix-SID of 10 for interface Loopback 0 with IPv4 address 1.0.0.1/32 results in assigning the label 9000010 16010 to the prefix 1.0.0.1/32.

BGP Prefix Segment Identifiers

Segments associated with a BGP prefix are known as BGP Prefix-SIDs.

- BGP Prefix-SIDs are always global within a Segment Routing or BGP domain
- BGP Prefix-SIDs identifies an instruction to forward the packet over ECMP-aware best path computed by BGP for a given prefix

Segment Routing requires BGP speaker to be configured with a Segment Routing Global block (SRGB). Generally, SRGB is configured as a range of labels, SRGB = [SR_S, SR_E].

- SR_S = Start of the range
- SR_E = End of the range

Each prefix is assigned with its own unique label index.

In the following example, a BGP route policy, set label index, is defined using the route-policy **name** command.

Configure the Segment Routing Global Block (SRGB) in BGP. If the route label path has a label-index attribute and SRGB is configured, then local label route is allocated from SRGB. If label-index is added to redistributed routes using route-policy, then BGP presents label-index as an attribute with the route.

```
router bgp 100
 bgp log-neighbor-changes
 neighbor 192.0.2.1 remote-as 100
 neighbor 192.0.2.1 update-source Loopback0
 neighbor 192.0.23.3 remote-as 300
 !
 address-family ipv4
  segment-routing mpls
  neighbor 192.0.2.1 activate
  neighbor 192.0.2.1 send-label
  neighbor 192.0.23.3 activate
 exit-address-family
```

Additional References for Segment Routing

Related Documents

Related Topic	Document Title
Videos	• Introduction to Cisco Segment Routing (YouTube) • Introduction to Cisco Segment Routing (CCO)

CHAPTER 2

Segment Routing—IS-IS v4 node SID

The Segment Routing—ISIS v4 node SID feature provides support for segment routing on Cisco Intermediate System-to-Intermediate System (IS-IS) networks.

Information About Segment Routing IS-IS v4 Node SID

Segment Routing IS-IS v4 Node SID

Segment Routing (SR) is a technique by which the path followed by a packet is encoded in the packet itself. The path followed by a packet in segment routing is encoded in the packet itself similar to loose or strict source routing. Segment Routing relies on a small number of extensions to Cisco Intermediate System-to-Intermediate System (IS-IS) and Open Shortest Path First (OSPF) protocols.

How to Configure Segment Routing —IS-IS v4 Node SID

Configuring Segment Routing

Before You Begin

Before configuring IS-IS to support segment routing you must first configure the segment routing feature in global configuration mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. segment-routing mpls
4. connected-prefix-sid-map
5. address-family ipv4
6. 1.1.1.1/32 index 100 range 1
7. exit-address-family

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	segment-routing mpls **Example:** `Device(config-sr)# segment-routing mpls`	Enables the segment feature using the mpls data plane.
Step 4	connected-prefix-sid-map **Example:** `Device(config-srmpls)# connected-prefix-sid-map`	Enters a sub-mode where you can configure address-family specific mappings for local prefixes and SIDs.
Step 5	address-family ipv4 **Example:** `Device(config-srmpls-conn)# address-family ipv4`	Specifies IPv4 address prefixes.
Step 6	1.1.1.1/32 index 100 range 1 **Example:** `Device(config-srmpls-conn-af)# 1.1.1.1/32 100 range 1`	Associates SID 100 with the address 1.1.1.1/32.

	Command or Action	Purpose
Step 7	exit-address-family **Example:** `Device(config-srmpls-conn-af)# exit-address-family`	Exits the address family.

Configuring Segment Routing on IS-IS Network

Before You Begin

Before you configure segment routing on IS-IS network, IS-IS must be enabled on your network.

SUMMARY STEPS

1. router isis
2. net network-entity-title
3. metric-style wide
4. **segment-routing** mpls
5. exit
6. show isis segment-routing

DETAILED STEPS

	Command or Action	Purpose
Step 1	router isis **Example:** `Device(config-if)# router isis`	Enables the IS-IS routing protocol and enters router configuration mode.
Step 2	net network-entity-title **Example:** `Device(config-router)# net 49.0000.0000.0003.00`	Configures network entity titles (NETs) for the routing instance.
Step 3	metric-style wide **Example:** `Device(config-router)# metric-style wide`	Configures the device to generate and accept only wide link metrics.

	Command or Action	Purpose
Step 4	**segment-routing** mpls **Example:** `Device(config-router)# segment-routing mpls`	Configures segment routing operation state.
Step 5	exit **Example:** `Device(config-router)# exit`	Exits segment routing mode and returns to the configuration terminal mode.
Step 6	show isis segment-routing **Example:** `Device# show is-is segment-routing`	(Optional) Displays the current state of the IS-IS segment routing.

The following example displays output from the show isis segment-routing state command for the segment routing under IS-IS:

```
Device# show isis segment-routing

ISIS protocol is registered with MFI
ISIS MFI Client ID:0x63
Tag 1 - Segment-Routing:
   SR State:SR_ENABLED
   Number of SRGB:1
   SRGB Start:16000, Range:8000, srgb_handle:0x4500AED0, srgb_state: created
   Address-family IPv4 unicast SR is configured
     Operational state:Enabled
```

Configuring Prefix-SID for IS-IS

This task explains how to configure prefix segment identifier (SID) index under each interface.

Before You Begin

Segment routing must be enabled on the corresponding address family.

SUMMARY STEPS

1. enable
2. configure terminal
3. segment-routing mpls
4. connected-prefix-sid-map
5. address-family ipv4
6. 1.1.1.1/32 index 100 range 1
7. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** `Device# enable`	Enables privileged EXEC mode.
Step 2	configure terminal **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	segment-routing mpls **Example:** `Device(config)# segment-routing mpls`	Configures segment routing mpls mode.
Step 4	connected-prefix-sid-map **Example:** `Device(config-srmpls)# connected-prefix-sid-map`	Enters a sub-mode where you can configure address-family specific mappings for local prefixes and SIDs.
Step 5	address-family ipv4 **Example:** `Device(config-srmpls-conn)# address-family ipv4`	Specifies the IPv4 address family and enters router address family configuration mode.
Step 6	1.1.1.1/32 index 100 range 1 **Example:** `Device(config-srmpls-conn-af)# 1.1.1.1/32 100 range 1`	Associates SID 100 with the address 1.1.1.1/32.
Step 7	exit **Example:** `Device(config-router)# exit`	Exits segment routing mode and returns to the configuration terminal mode.

Configuring Prefix Attribute N-flag-clear

By default, a flag called N-flag is set by IS-IS when advertising a SID which is associated with a loopback address. If you wish to clear this flag add explicit configuration.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. interface loopback3
4. isis prefix n-flag-clear

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	interface loopback3 **Example:** `Device(config)# interface loopback3`	Specifies the interface loopback.
Step 4	isis prefix n-flag-clear **Example:** `Device(config-if)# isis prefix n-flag-clear`	Clears the prefix N-flag.

Configuring Explicit Null Attribute

To disable penultimate-hop-popping (PHP) and add explicit-Null label, explicit-null option needs to be specified. Once the option is given, IS-IS sets the E flag in the prefix-SID sub TLV.

By default, a flag called E-flag (Explicit-Null flag) is set to 0 by ISIS when advertising a Prefix SID which is associated with a loopback address. If you wish to set this flag add explicit configuration

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. segment-routing mpls
4. set-attributes
5. address-family ipv4
6. explicit-null
7. exit-address-family

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	segment-routing mpls **Example:** `Device(config)# segment-routing mpls`	Configures segment routing mpls mode.
Step 4	set-attributes **Example:** `Device(config-srmpls)# set-attributes`	Sets the attribute.
Step 5	address-family ipv4 **Example:** `Device(config-srmpls-attr)# address-family ipv4`	Specifies the IPv4 address family and enters router address family configuration mode.
Step 6	explicit-null **Example:** `Device(config-srmpls-attr-af)# explicit-null`	Specifies the explicit-null.

	Command or Action	Purpose
Step 7	exit-address-family **Example:** `Device(config-srmpls-attr-af)# exit-address-family`	Exits the address family.

Configuration Examples for Segment Routing —IS-IS v4 Node SID

Example: Configuring Segment Routing on IS-IS Network

The following example shows how to configure prefix segment identifier (SID) index under each interface:

```
Device(config)#segment-routing mpls
 Device(config-srmpls)#connected-prefix-sid-map
  Device(config-srmpls-conn)#address-family ipv4
   Device(config-srmpls-conn-af)#10.1.2.2/32 index 2 range 1
  Device(config-srmpls-conn-af)#exit-address-family
 Device(config-srmpls-conn-af)#end
```

Example: Configuring Explicit Null Attribute

The following is an example for configuring explicit null attribute:

```
Device(config)# segment-routing mpls
Device(config-srmpls)# set-attributes
 Device(config-srmpls-attr)# address-family ipv4
  Device(config-srmpls-attr-af)# explicit-null
 Device (config-srmpls-attr-af)# exit-address-family
```

Additional References for Segment Routing-IS-IS v4 Node SID

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/mcl/allreleasemcl/all-book.html

Related Topic	Document Title
IP Routing ISIS commands	Cisco IOS IP Routing ISIS commands http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/mcl/allreleasemcl/all-book.html

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Segment Routing—IS-IS v4 Node SID

Table 1: Feature Information for Segment Routing—IS-IS v4 Node SID

Feature Name	Releases	Feature Information
Segment Routing—IS-IS v4 Node SID	Cisco IOS XE Release 3.16S	The Segment Routing—ISIS v4 node SID feature provides support for segment routing on IS-IS networks. The following commands were introduced or modified: **connected-prefix-sid-map**, **show isis segment-routing**, **isis prefix n-flag-clear**, **explicit-null**

CHAPTER 3

Segment Routing OAM Support

This chapter describes how to verify the operation with Segment Routing OAM label switched protocol ping and traceroute (SR OAM LSPV).

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

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

Restrictions for Segment Routing OAM Support

- Ping and traceroute does not display proper output over SR-TE tunnels using verbatim path option.
- Ping and traceroute are unsupported with SR-TE Static auto tunnel, BGP Dynamic TE, and On-demand next hop auto tunnels.

Information About Segment Routing-OAM Support

Segment Routing-OAM Support

The Segment Routing-OAM Support feature provides support for Nil-FEC LSP Ping/Trace functionality. Nil-FEC LSP Ping/Trace functionality support Segment Routing and MPLS Static. It also act as an additional diagnostic tool for all other LSP types. This feature allows operators to provide the ability to freely test any label stack by allowing them to specify the following:

- label stack
- outgoing interface
- nexthop address

In the case of segment routing, each segment nodal label and adjacent label along the routing path is put into the label stack of an echo request message from initiator Label Switch Router (LSR); MPLS data plane forward this packet to the label stack target, and the label stack target reply the echo message back.

LSP Ping Operation for Nil FEC target

The LSP Ping/Traceroute is used in identifying LSP breakages. The nil-fec target type can be used to test the connectivity for a known label stack. Follow the existing LSP ping procedure (for more information, refer MPLS LSP Ping/Traceroute), with the following modifications:

- Build the echo request packet with the given label stack;
- Append explicit null label at the bottom of the label stack;
- Build echo request FTS TLV with target FEC Nil FEC and label value set to the bottom label of the label stack, which is explicit-null.

How to Diagnose Segment Routing with LSP Ping and Trace Route Nil FEC Target

Use LSP Ping for Nil FEC Target

The Nil-FEC LSP ping and traceroute operation are simply extension of regular MPLS ping and trace route. nil-fec labels <label, label…> is added to the ping mpls command. This command sends an echo request message with MPLS label stack as specified, and add another explicit null at bottom of the stack.

```
ping mpls
   {{ipv4 <target>/<mask> [fec-type {bgp | generic | ldp}] |
    {pseudowire <peer addr> <vc-id> [segment<segment-number>]} |
    {traffic-eng {<tunnel interface> |
          {p2p <sender> <endpoint>
    <tun-id> <ex-tun-id> <lspid>} |
          {p2mp <p2mp-id> <sender>
     <tun-id><ex-tun-id> <lspid>}} |
```

```
     {tp <tunnel-tp interface> lsp <working | protect | active>}} |
     {mldp {p2mp | mp2mp}
   {{ipv4 <source> <group>} |
    {ipv6 <source> <group>} |
    {vpnv4 <rd> <source> <group>} |
    {vpnv6 <rd> <source> <group>} |
    {hex <opaque type> <opaque value>} |
     {mdt <vpnid> <mdt-num>} |
    {static-id <lsp-identifier>} |
    {global-id <global-identifier>}}} |
    {nil-fec {labels <comma separated labels>}}
    [repeat <count>]
    [size <size> | sweep <min_size> <max_size> <increment>]
    [timeout <seconds>]
    [interval <milliseconds>]
    [destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
    [source <addr>]
    [exp <exp-value>]
    [pad <pattern>]
    [ttl <ttl>]
    [reply [mode [ipv4 | router-alert | no-reply]]
    [dscp <dscp-bits>]
    [pad-tlv]]
    [verbose]
    [force-explicit-null]
    [force-disposition ra-label]
    [output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
   [{dsmap | ddmap [l2ecmp]} [hashkey {none | {ipv4 | ipv4-label-set {bitmap <bitmap_size>}}]

    [flags {fec | ttl}]
    [segment {all | <lower-segment-number> [upper-segment-number]}
    [jitter <milliseconds>]
    [responder-id <ip addr> | egress <ipaddr>]
      NOTE: responder-id for mldp/ egress for p2mp te
    [entropy-label <label-value>]
```

For more information, refer ping mpls.

Use LSP Traceroute for Nil FEC Target

```
trace mpls
    {{ipv4 <target>/<mask> [fec-type {bgp | generic | ldp}]} |
     {traffic-eng {<tunnel interface> |
           {p2p <sender> <endpoint>
        <tun-id> <ex-tun-id> <lspid>} |
       {p2mp <p2mp-id> <sender>
      <tun-id><ex-tun-id> <lspid>}} |
     {mldp {p2mp | mp2mp}
   {{ip <source> <group>} |
    {vpn <rd> [<source> <group>]} |
    {hex <opaque type> <opaque value>} |
     {mdt <vpnid> <mdt-num>} |
    {static-id <lsp-identifier>} |
    {global-id <global-identifier>}}}
     {pseudowire <next-pe-address> <pwid> [segment
      <segment-number/lower-segment-number>[<upper-segment-limit>]]} |
     {tp <tunnel-tp interface> lsp <working | protect | active>}} |
    {nil-fec {labels <comma separated labels>}}
    [timeout <seconds>]
    [destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
    [source <addr> ]
    [exp <exp-value>]
    [ttl <ttl-max>]
    [reply [mode [ipv4 | router-alert | no-reply]]
    [dscp <dscp-bits>]
    [pad-tlv]]
    [force-explicit-null]
    [output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
    [flags {fec | ttl}]
    [entropy-label <label-value>]
```

```
[hashkey ipv4 | ipv4-label-set {bitmap <bitmap_size>}]
```

For more information, refer to the traceroute mpls.

Example: LSP Ping Nil_FEC Target Support

```
Node loopback IP address: 1.1.1.3              1.1.1.4                1.1.1.5
     1.1.1.7
Node label:                                     16004                  16005
      16007
Nodes:                    Arizona --------------- Utah  --------------- Wyoming
--------------- Texas
Interface:                     Eth1/0        Eth1/0
Interface IP address:         30.1.1.3      30.1.1.4

SR232-utah#sh mpls forwarding-table
Local      Outgoing    Prefix           Bytes Label   Outgoing   Next Hop
Label      Label       or Tunnel Id     Switched      interface
16         Pop Label   3333.3333.0000-Et1/0-30.1.1.3   \
                                        0             Et1/0      30.1.1.3
17         Pop Label   5555.5555.5555-Et1/1-90.1.1.5   \
                                        0             Et1/1      90.1.1.5
18         Pop Label   3333.3333.0253-Et0/2-102.102.102.2   \
                                        0             Et0/2      102.102.102.2
19         Pop Label   9.9.9.4/32       0             Et0/2      102.102.102.2
20         Pop Label   1.1.1.5/32       0             Et1/1      90.1.1.5
21         Pop Label   1.1.1.3/32       0             Et1/0      30.1.1.3
22         Pop Label   16.16.16.16/32   0             Et1/0      30.1.1.3
23         Pop Label   16.16.16.17/32   0             Et1/0      30.1.1.3
24         Pop Label   17.17.17.17/32   0             Et1/0      30.1.1.3
25         20          9.9.9.3/32       0             Et1/0      30.1.1.3
26         21          1.1.1.6/32       0             Et1/0      30.1.1.3
27         24          1.1.1.2/32       0             Et1/0      30.1.1.3
           28          1.1.1.2/32       0             Et1/1      90.1.1.5
28         18          1.1.1.7/32       0             Et1/1      90.1.1.5
29         27          9.9.9.7/32       0             Et1/1      90.1.1.5
30         Pop Label   55.1.1.0/24      0             Et1/1      90.1.1.5
31         Pop Label   19.1.1.0/24      0             Et1/0      30.1.1.3
Local      Outgoing    Prefix           Bytes Label   Outgoing   Next Hop
Label      Label       or Tunnel Id     Switched      interface
32         Pop Label   100.1.1.0/24     0             Et1/0      30.1.1.3
33         Pop Label   100.100.100.0/24 0             Et1/0      30.1.1.3
34         Pop Label   110.1.1.0/24     0             Et1/0      30.1.1.3
35         28          10.1.1.0/24      0             Et1/0      30.1.1.3
36         29          101.101.101.0/24 0             Et1/0      30.1.1.3
37         29          65.1.1.0/24      0             Et1/1      90.1.1.5
38         33          104.104.104.0/24 0             Et1/0      30.1.1.3
           39          104.104.104.0/24 0             Et1/1      90.1.1.5
39         30          103.103.103.0/24 0             Et1/1      90.1.1.5
16005      Pop Label   1.1.1.5/32       1782          Et1/1      90.1.1.5
16006      16006       1.1.1.6/32       0             Et1/0      30.1.1.3
16007      16007       1.1.1.7/32       0             Et1/1      90.1.1.5
16017      16017       17.17.17.17/32   0             Et1/0      30.1.1.3
16250      16250       9.9.9.3/32       0             Et1/0      30.1.1.3
16252      16252       9.9.9.7/32       0             Et1/1      90.1.1.5
16253      Pop Label   9.9.9.4/32       0             Et0/2      102.102.102.2
17000      17000       16.16.16.16/32   0             Et1/0      30.1.1.3
17002      17002       1.1.1.2/32       0             Et1/0      30.1.1.3
           17002       1.1.1.2/32       0             Et1/1      90.1.1.5

SR231-arizona#ping mpls nil-fec labels 16005,16007 output interface ethernet 1/0 nexthop
30.1.1.4 repeat 1
Sending 1, 72-byte MPLS Echos with Nil FEC labels 16005,16007,
     timeout is 2 seconds, send interval is 0 msec:

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
```

```
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 1/1/1 ms
 Total Time Elapsed 0 ms


SR231-arizona#traceroute mpls nil-fec labels 16005,16007 output interface ethernet 1/0
nexthop 30.1.1.4
Tracing MPLS Label Switched Path with Nil FEC labels 16005,16007, timeout is 2 seconds

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
  0 30.1.1.3 MRU 1500 [Labels: 16005/16007/explicit-null Exp: 0/0/0]
L 1 30.1.1.4 MRU 1500 [Labels: implicit-null/16007/explicit-null Exp: 0/0/0] 1 ms
L 2 90.1.1.5 MRU 1500 [Labels: implicit-null/explicit-null Exp: 0/0] 1 ms
! 3 55.1.1.7 1 ms
```

Additional References for Segment Routing-OAM Support

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Segment Routing-OAM Support

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 2: Feature Information for Segment Routing-OAM Support

Feature Name	Releases	Feature Information
Segment Routing-OAM Support	Cisco IOS XE Release 3.17 S	The Segment Routing-OAM Support feature provides support for Nil-FEC LSP Ping/Trace functionality.