CONTENTS

PREFACE

Preface vii
Changes to This Document vii
Communications, Services, and Additional Information vii

CHAPTER 1
New and Changed Information for Segment Routing Features 1
New and Changed Segment Routing Features 1

CHAPTER 2
About Segment Routing 3
Scope 3
Need 4
Benefits 4
Workflow for Deploying Segment Routing 4

CHAPTER 3
Configure Segment Routing Global Block 7
About the Segment Routing Global Block 7
Setup a Non-Default Segment Routing Global Block Range 8

CHAPTER 4
Configure Segment Routing for IS-IS Protocol 11
Enabling Segment Routing for IS-IS Protocol 11
Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface 13
IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability 14
Prefix Attribute Flags 15
IPv4 and IPv6 Source Router ID 16
Configuring Prefix Attribute N-flag-clear 17
IS-IS Multi-Domain Prefix SID and Domain Stitching: Example 18
Configure IS-IS Multi-Domain Prefix SID 18
MPLS LSP Ping and Traceroute Nil FEC Target 47
Examples: LSP Ping and Traceroute for Nil_FEC Target 48
Preface

The Segment Routing Configuration Guide for Cisco NCS 5000 Series Routers preface contains these sections:

- Changes to This Document, on page vii
- Communications, Services, and Additional Information, on page vii

Changes to This Document

This table lists the changes made to this document since it was first printed.

<table>
<thead>
<tr>
<th>Date</th>
<th>Change Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>March 2018</td>
<td>Initial release of this document</td>
</tr>
</tbody>
</table>

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you’re looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.
New and Changed Information for Segment Routing Features

This table summarizes the new and changed feature information for the Segment Routing Configuration Guide for Cisco NCS 5000 Series Routers, and lists where they are documented.

- New and Changed Segment Routing Features, on page 1

### New and Changed Segment Routing Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Introduced/Changed in Release</th>
<th>Where Documented</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>No new features introduced.</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>
About Segment Routing

This chapter introduces the concept of segment routing and provides a workflow for configuring segment routing.

- Scope, on page 3
- Need, on page 4
- Benefits, on page 4
- Workflow for Deploying Segment Routing, on page 4

Scope

Segment routing is a method of forwarding packets on the network based on the source routing paradigm. The source chooses a path and encodes it in the packet header as an ordered list of segments. Segments are an identifier for any type of instruction. For example, topology segments identify the next hop toward a destination. Each segment is identified by the segment ID (SID) consisting of a flat unsigned 20-bit integer.

Segments

Interior gateway protocol (IGP) distributes two types of segments: prefix segments and adjacency segments. Each router (node) and each link (adjacency) has an associated segment identifier (SID).

- A prefix SID is associated with an IP prefix. The prefix SID is manually configured from the segment routing global block (SRGB) range of labels, and is distributed by IS-IS or OSPF. The prefix segment steers the traffic along the shortest path to its destination. A node SID is a special type of prefix SID that identifies a specific node. It is configured under the loopback interface with the loopback address of the node as the prefix.

  A prefix segment is a global segment, so a prefix SID is globally unique within the segment routing domain.

- An adjacency segment is identified by a label called an adjacency SID, which represents a specific adjacency, such as egress interface, to a neighboring router. The adjacency SID is distributed by IS-IS or OSPF. The adjacency segment steers the traffic to a specific adjacency.

  An adjacency segment is a local segment, so the adjacency SID is locally unique relative to a specific router.

By combining prefix (node) and adjacency segment IDs in an ordered list, any path within a network can be constructed. At each hop, the top segment is used to identify the next hop. Segments are stacked in order at
the top of the packet header. When the top segment contains the identity of another node, the receiving node uses equal cost multipaths (ECMP) to move the packet to the next hop. When the identity is that of the receiving node, the node pops the top segment and performs the task required by the next segment.

**Dataplane**

Segment routing can be directly applied to the Multiprotocol Label Switching (MPLS) architecture with no change in the forwarding plane. 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.

**Services**

Segment Routing 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).

**Need**

With segment routing for traffic engineering (SR-TE), the network no longer needs to maintain a per-application and per-flow state. Instead, it simply obeys the forwarding instructions provided in the packet.

SR-TE utilizes network bandwidth more effectively than traditional MPLS-TE networks by using ECMP at every segment level. It uses a single intelligent source and relieves remaining routers from the task of calculating the required path through the network.

**Benefits**

- **Ready for SDN**: Segment routing was built for SDN and is the foundation for Application Engineered Routing (AER). SR prepares networks for business models, where applications can direct network behavior. SR provides the right balance between distributed intelligence and centralized optimization and programming.
- **Minimal configuration**: Segment routing for TE requires minimal configuration on the source router.
- **Load balancing**: Unlike in RSVP-TE, load balancing for segment routing can take place in the presence of equal cost multiple paths (ECMPs).
- **Supports Fast Reroute (FRR)**: Fast reroute enables the activation of a pre-configured backup path within 50 milliseconds of path failure.
- **Plug-and-Play deployment**: Segment routing are interoperable with existing MPLS control and data planes and can be implemented in an existing deployment.

**Workflow for Deploying Segment Routing**

Follow this workflow to deploy segment routing.

1. Configure the Segment Routing Global Block (SRGB)
2. Enable Segment Routing and Node SID on the IGP
3. Configure Segment Routing on the BGP
4. Configure the Segment Routing Mapping Server
Configure Segment Routing Global Block

Local label allocation is managed by the label switching database (LSD). The Segment Routing Global Block (SRGB) is the range of label values preserved for segment routing in the LSD.

• About the Segment Routing Global Block, on page 7
• Setup a Non-Default Segment Routing Global Block Range, on page 8

About the Segment Routing Global Block

The SRGB label values are assigned as prefix segment identifiers (SIDs) to SR-enabled nodes and have global significance throughout the domain.

Note
Because the values assigned from the range have domain-wide significance, we recommend that all routers within the domain be configured with the same range of values.

The default SRGB range is from 16000 to 23999.

Note
On SR-capable routers, the default starting value of the dynamic label range is increased from 16000 to 24000, so that the default SRGB label values (16000 to 23999) are available when SR is enabled on a running system. If a dynamic label range has been configured with a starting value of 16000, then the default SRGB label values may already be in use when SR is enabled on a running system. Therefore, you must reload the router after enabling SR to release the currently allocated labels and allocate the SRGB.

Also, if you need to increase the SRGB range after you have enabled SR, you must reload the router to release the currently allocated labels and allocate the new SRGB.

To keep the segment routing configuration simple and to make it easier to troubleshoot segment routing issues, we recommend that you use the default SRGB range on each node in the domain. However, there are instances when you might need to define a different range. For example:

• The nodes of another vendor support a label range that is different from the default SRGB, and you want to use the same SRGB on all nodes.

• The default range is too small.

• To specify separate SRGBs for IS-IS and OSPF protocols, as long as the ranges do not overlap.
Restrictions:

• In Cisco IOS XR release 6.2.x and earlier, LSD label values 0-15999 are reserved. In Cisco IOS XR release 6.3.1 and later, LSD label values 0-14999 are reserved.

• In Cisco IOS XR release 6.2.x and earlier, the maximum SRGB size is 65536. In Cisco IOS XR release 6.3.1 and later, the maximum SRGB size is 262,143.

• The SRGB upper bound cannot exceed the platform's capability.

Note
The NCS 5001 and NCS 5002 support a total of 16000 labels. The NCS 5011 supports a total of 13800 labels. Although you can configure the SRGB to any range, an Out of Resource (OOR) mechanism in hardware prevents the platform from programming more labels. When an OOR condition occurs, reduce the label scale and reload the router.

Note
Label values that are not previously reserved are available for dynamic assignment.

The SRGB can be disabled if SR is not used.

Setup a Non-Default Segment Routing Global Block Range

This task explains how to configure a non-default SRGB range.

SUMMARY STEPS

1. configure
2. [router {isis instance-id | ospf process_name} ]
3. segment-routing global-block starting_value ending_value
4. commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 configure</td>
<td></td>
</tr>
<tr>
<td>Step 2 [router {isis instance-id</td>
<td>ospf process_name} ]</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config)# router isis 1</td>
<td></td>
</tr>
<tr>
<td>Step 3 segment-routing global-block starting_value ending_value</td>
<td>Enter the lowest value that you want the SRGB range to include as the starting value. Enter the highest value that you want the SRGB range to include as the ending value.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config-isis)# segment-routing</td>
<td></td>
</tr>
</tbody>
</table>
Verify the SRGB configuration:

```
RP/0/RP0/CPU0:router# show mpls label table detail
Table Label Owner State Rewrite
----- ------- ------------------------------- ------ -------
<...snip...>
0 18000 ISIS(A):1 InUse No
   Lbl-blk SRGB, vers:0, (start_label=18000, size=2000)
0 24000 ISIS(A):1 InUse Yes
   (SR Adj Segment IPv4, vers:0, index-1, type-0, intf=Gi0/0/0/0, nh=10.0.0.2)
```

**What to do next**

Configure prefix SIDs and enable segment routing.
Setup a Non-Default Segment Routing Global Block Range
CHAPTER 4

Configure Segment Routing for IS-IS Protocol


This module provides the configuration information used to enable segment routing for IS-IS.

For additional information on implementing IS-IS on your Cisco NCS 5000 Series Router, see the Implementing IS-IS module in the Routing Configuration Guide for Cisco NCS 5000 Series Routers.

Note

- Enabling Segment Routing for IS-IS Protocol, on page 11
- Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface, on page 13
- IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability, on page 14
- IS-IS Multi-Domain Prefix SID and Domain Stitching: Example, on page 18

Enabling Segment Routing for IS-IS Protocol

Segment routing on the IS-IS control plane supports the following:

- IPv4 control plane
- Level 1, level 2, and multi-level routing
- Prefix SIDs for host prefixes on loopback interfaces
- Adjacency SIDs for adjacencies
- MPLS penultimate hop popping (PHP) and explicit-null signaling

This task explains how to enable segment routing for IS-IS.

Before you begin

Your network must support the MPLS Cisco IOS XR software feature before you enable segment routing for IS-IS on your router.
You must enter the commands in the following task list on every IS-IS router in the traffic-engineered portion of your network.

### SUMMARY STEPS

1. configure
2. router isis instance-id
3. address-family ipv4 [ unicast ]
4. metric-style wide [ level { 1 | 2 }]
5. segment-routing mpls
6. exit
7. commit

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 configure</td>
<td>Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.</td>
</tr>
<tr>
<td>Step 2 router isis instance-id</td>
<td>You can change the level of routing to be performed by a particular routing instance by using the is-type router configuration command.</td>
</tr>
<tr>
<td>Example: RP/0/RP0/CPU0:router(config)# router isis isp</td>
<td></td>
</tr>
<tr>
<td>Step 3 address-family ipv4 [ unicast ]</td>
<td>Specifies the IPv4 address family, and enters router address family configuration mode.</td>
</tr>
<tr>
<td>Example: RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast</td>
<td></td>
</tr>
<tr>
<td>Step 4 metric-style wide [ level { 1</td>
<td>2 }]</td>
</tr>
<tr>
<td>Example: RP/0/RP0/CPU0:router(config-isis-af)# metric-style wide level 1</td>
<td></td>
</tr>
<tr>
<td>Step 5 segment-routing mpls</td>
<td>Segment routing is enabled by the following actions:</td>
</tr>
<tr>
<td>Example: RP/0/RP0/CPU0:router(config-isis-af)# segment-routing mpls</td>
<td>• MPLS forwarding is enabled on all interfaces where IS-IS is active.</td>
</tr>
<tr>
<td></td>
<td>• All known prefix-SIDs in the forwarding plain are programmed, with the prefix-SIDs advertised by remote routers or learned through local or remote mapping server.</td>
</tr>
<tr>
<td></td>
<td>• The prefix-SIDs locally configured are advertised.</td>
</tr>
</tbody>
</table>
Example:

RP/0/RP0/CPU0:router(config-isis-af)# exit
RP/0/RP0/CPU0:router(config-isis)# exit

Step 7  commit

What to do next
Configure the prefix SID.

Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface

A prefix segment identifier (SID) is associated with an IP prefix. The prefix SID is manually configured from the segment routing global block (SRGB) range of labels. A prefix SID is configured under the loopback interface with the loopback address of the node as the prefix. The prefix segment steers the traffic along the shortest path to its destination.

A prefix SID can be a node SID or an Anycast SID. A node SID is a type of prefix SID that identifies a specific node. An Anycast SID is a type of prefix SID that identifies a set of nodes, and is configured with n-flag clear. The set of nodes (Anycast group) is configured to advertise a shared prefix address and prefix SID. Anycast routing enables the steering of traffic toward multiple advertising nodes. Packets addressed to an Anycast address are forwarded to the topologically nearest nodes.

The prefix SID is globally unique within the segment routing domain.

This task explains how to configure prefix segment identifier (SID) index or absolute value on the IS-IS enabled Loopback interface.

Before you begin
Ensure that segment routing is enabled on the corresponding address family.

SUMMARY STEPS

1. configure
2. router isis instance-id
3. interface Loopback instance
4. address-family ipv4 [ unicast ]
5. commit

DETAILED STEPS
### IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability

The following sub-TLVs support the advertisement of IPv4 and IPv6 prefix attribute flags and the source router ID of the router that originated a prefix advertisement, as described in RFC 7794.

- Prefix Attribute Flags
- IPv4 and IPv6 Source Router ID
Prefix Attribute Flags

The Prefix Attribute Flag sub-TLV supports the advertisement of attribute flags associated with prefix advertisements. Knowing if an advertised prefix is directly connected to the advertising router helps to determine how labels that are associated with an incoming packet should be processed.

This section describes the behavior of each flag when a prefix advertisement is learned from one level to another.

Prefix attributes are only added when wide metric is used.

Prefix Attribute Flags Sub-TLV Format

```
0 1 2 3 4 5 6 7 ... 
+---------------+...
|X|R|N| ... 
+---------------+...
```

Prefix Attribute Flags Sub-TLV Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X (External Prefix Flag)</td>
<td>This flag is set if the prefix has been redistributed from another protocol. The value of the flag is preserved when the prefix is propagated to another level.</td>
</tr>
<tr>
<td>R (Re-advertisement Flag)</td>
<td>This flag is set to 1 by the Level 1-2 router when the prefix is propagated between IS-IS levels (from Level 1 to Level 2, or from Level 2 to Level 1). This flag is set to 0 when the prefix is connected locally to an IS-IS-enabled interface (regardless of the level configured on the interface).</td>
</tr>
</tbody>
</table>
For prefixes that are propagated from another level:

1. Copy the N-flag from the prefix attribute sub-TLV, if present in the source level.
2. Copy the N-flag from the prefix-SID sub-TLV, if present in the source level.
3. Otherwise, set to 0.

For connected prefixes:

1. Set to 0 if `prefix-attributes n-flag-clear` is configured (see Configuring Prefix Attribute N-flag-clear).
2. Set to 0 if `n-flag-clear { n-flag-clearSID-index | n-flag-clearSID-value }` is configured (see Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface).
3. Otherwise, set to 1 when the prefix is a host prefix (/32 for IPV4, /128 for IPv6) that is associated with a loopback address.

**Note** If the flag is set and the prefix length is not a host prefix, then the flag must be ignored.

### IPv4 and IPv6 Source Router ID

The Source Router ID sub-TLV identifies the source of the prefix advertisement. The IPv4 and IPv6 source router ID is displayed in the output of the `show isis database verbose` command.

The Source Router ID sub-TLV is added when the following conditions are met:

1. The prefix is locally connected.
2. The N-flag is set to 1 (when it's a host prefix and the `n-flag-clear` configuration is not used).
3. The router ID is configured in the corresponding address family.

The source router ID is propagated between levels.

**Table 1: Source Router Sub-TLV Format**

| IPv4 Source Router ID | Type: 11  
|-----------------------|-----------------------  
| Length: 4  
| Value: IPv4 Router ID of the source of the prefix advertisement |

| IPv6 Source Router ID | Type: 12  
|-----------------------|
| Length: 16  
| Value: IPv6 Router ID of the source of the prefix advertisement |
Configuring Prefix Attribute N-flag-clear

The N-flag is set to 1 when the prefix is a host prefix (/32 for IPV4, /128 for IPv6) that is associated with a loopback address. The advertising router can be configured to not set this flag. This task explains how to clear the N-flag.

SUMMARY STEPS

1. configure
2. router isis instance-id
3. interface Loopback instance
4. prefix-attributes n-flag-clear [Level-1 | Level-2]
5. commit

DETAIL STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 configure</td>
<td></td>
</tr>
<tr>
<td>Step 2 router isis instance-id</td>
<td>Specifies the loopback interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config)# router isis 1</td>
<td></td>
</tr>
<tr>
<td>Step 3 interface Loopback instance</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config)# interface Loopback0</td>
<td></td>
</tr>
<tr>
<td>Step 4 prefix-attributes n-flag-clear [Level-1</td>
<td>Level-2]</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config-if)# isis prefix-attributes n-flag-clear</td>
<td></td>
</tr>
<tr>
<td>Step 5 commit</td>
<td></td>
</tr>
</tbody>
</table>

Verify the prefix attribute configuration:

```
RP/0/RP0/CPU0:router# show isis database verbose
```

```
IS-IS 1 (Level-2) Link State Database
LSPID        LSP Seq Num LSP Checksum LSP Holdtime ATT/P/OL
router.00-00 * 0x0000039b 0xfc27 1079 0/0/0
Area Address: 49.0001
NLPID:       0xcc
NLPID:       0x8e
MT:          Standard (IPv4 Unicast)
MT:          IPv6 Unicast
Hostname:    router
IP Address:  10.0.0.1
IPv6 Address: 2001:0db8:1234::0a00:0001
```

Segment Routing Configuration Guide for Cisco NCS 5000 Series Routers, IOS XR Release 6.4.x
IS-IS Multi-Domain Prefix SID and Domain Stitching: Example

IS-IS Multi-Domain Prefix SID and Domain Stitching allows you to configure multiple IS-IS instances on the same loopback interface for domain border nodes. You specify a loopback interface and prefix SID under multiple IS-IS instances to make the prefix and prefix SID reachable in different domains.

This example uses the following topology. Node 5 and 9 are border nodes between two IS-IS domains (Domain1 and Domain2). Node 10 is configured as the Segment Routing Path Computation Element (SR-PCE).

*Figure 1: Multi-Domain Topology*

Configure IS-IS Multi-Domain Prefix SID

Specify a loopback interface and prefix SID under multiple IS-IS instances on each border node:

*Example: Border Node 5*
Border nodes 5 and 9 each run two IS-IS instances (Domain1 and Domain2) and advertise their Loopback0 prefix and prefix SID in both domains.

Nodes in both domains can reach the border nodes by using the same prefix and prefix SID. For example, Node 3 and Node 22 can reach Node 5 using prefix SID 16005.

### Configure Common Router ID

On each border node, configure a common TE router ID under each IS-IS instance:

**Example: Border Node 5**

```text
router isis Domain1
  address-family ipv4 unicast
  router-id loopback0

router isis Domain2
  address-family ipv4 unicast
  router-id loopback0
```

**Example: Border Node 9**

```text
router isis Domain1
  address-family ipv4 unicast
  router-id loopback0

router isis Domain2
  address-family ipv4 unicast
  router-id loopback0
```
Distribute IS-IS Link-State Data

Example: Node 13
router isis Domain1
    distribute link-state id

Example: Node 14
router isis Domain2
    distribute link-state id

Link-state ID starts from 32. One ID is required per IGP domain. Different domain IDs are essential to identify that the SR-TE TED belongs to a particular IGP domain.

Nodes 13 and 14 each reports its local domain in BGP-LS to Node 10.

Node 10 identifies the border nodes (Nodes 5 and 9) by their common advertised TE router ID, then combines (stitches) the domains on these border nodes for end-to-end path computations.
Configure Segment Routing for OSPF Protocol

Open Shortest Path First (OSPF) is an Interior Gateway Protocol (IGP) developed by the OSPF working group of the Internet Engineering Task Force (IETF). Designed expressly for IP networks, OSPF supports IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets.

This module provides the configuration information to enable segment routing for OSPF.

Note
For additional information on implementing OSPF on your router, see the Implementing OSPF module in the.

- Enabling Segment Routing for OSPF Protocol, on page 21
- Configuring a Prefix-SID on the OSPF-Enabled Loopback Interface, on page 23

Enabling Segment Routing for OSPF Protocol

Segment routing on the OSPF control plane supports the following:

- OSPFv2 control plane
- Multi-area
- IPv4 prefix SIDs for host prefixes on loopback interfaces
- Adjacency SIDs for adjacencies
- MPLS penultimate hop popping (PHP) and explicit-null signaling

This section describes how to enable segment routing MPLS and MPLS forwarding in OSPF. Segment routing can be configured at the instance, area, or interface level.

Before you begin
Your network must support the MPLS Cisco IOS XR software feature before you enable segment routing for OSPF on your router.
You must enter the commands in the following task list on every OSPF router in the traffic-engineered portion of your network.

### SUMMARY STEPS

1. configure
2. router ospf  process-name
3. segment-routing mpls
4. area 0
5. segment-routing mpls
6. exit
7. commit

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enables OSPF routing for the specified routing process and places the router in router configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong> router ospf  process-name</td>
<td>Enables segment routing using the MPLS data plane on the routing process and all areas and interfaces in the routing process.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Enables segment routing forwarding on all interfaces in the routing process and installs the SIDs received by OSPF in the forwarding table.</td>
</tr>
<tr>
<td><strong>Step 3</strong> segment-routing mpls</td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td><strong>Step 4</strong> area 0</td>
<td>Enters area configuration mode.</td>
</tr>
<tr>
<td><strong>Step 5</strong> segment-routing mpls</td>
<td>(Optional) Enables segment routing using the MPLS data plane on the area and all interfaces in the area. Enables segment routing forwarding on all interfaces in the area and installs the SIDs received by OSPF in the forwarding table.</td>
</tr>
<tr>
<td><strong>Step 6</strong> exit</td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td><strong>Step 7</strong> commit</td>
<td></td>
</tr>
</tbody>
</table>
What to do next
Configure the prefix SID.

Configuring a Prefix-SID on the OSPF-Enabled Loopback Interface

A prefix segment identifier (SID) is associated with an IP prefix. The prefix SID is manually configured from the segment routing global block (SRGB) range of labels. A prefix SID is configured under the loopback interface with the loopback address of the node as the prefix. The prefix segment steers the traffic along the shortest path to its destination.

A prefix SID can be a node SID or an Anycast SID. A node SID is a type of prefix SID that identifies a specific node. An Anycast SID is a type of prefix SID that identifies a set of nodes, and is configured with n-flag clear. The set of nodes (Anycast group) is configured to advertise a shared prefix address and prefix SID. Anycast routing enables the steering of traffic toward multiple advertising nodes. Packets addressed to an Anycast address are forwarded to the topologically nearest nodes.

The prefix SID is globally unique within the segment routing domain.

This task describes how to configure prefix segment identifier (SID) index or absolute value on the OSPF-enabled Loopback interface.

Before you begin
Ensure that segment routing is enabled on an instance, area, or interface.

SUMMARY STEPS

1. configure
2. router ospf  process-name
3. area  value
4. interface Loopback  interface-instance
5. commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>configure</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>router ospf  process-name</td>
<td>Enables OSPF routing for the specified routing process, and places the router in router configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RP0/CPU0:router(config)# router ospf 1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>area  value</td>
<td>Enters area configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RP/0/RP0/CPU0:router(config-ospf)# area 0</td>
<td></td>
</tr>
</tbody>
</table>
### Configuring a Prefix-SID on the OSPF-Enabled Loopback Interface

<table>
<thead>
<tr>
<th>Step 4</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 4** | **interface Loopback** **interface-instance**  
*Example:*  
RP/0/RP0/CPU0:router(config-ospf-ar)# **interface Loopback0 passive**  
|  | Specifies the loopback interface and instance. |

**Step 5**  
commit

Verify the prefix-SID configuration:

RP/0/RP0/CPU0:router# **show ospf database opaque-area 7.0.0.1 self-originate**  
OSPF Router with ID (10.0.0.1) (Process ID 1)  
Type-10 Opaque Link Area Link States (Area 0)  
  
  | Extended Prefix TLV: Length: 20  
  | Route-type: 1  
  | AF : 0  
  | Flags : 0x40  
  | Prefix : 10.0.0.1/32  

  | SID sub-TLV: Length: 8  
  | Flags : 0x0  
  | MTID : 0  
  | Algo : 0  

**SID Index : 1001**
CHAPTER 6

Configure Segment Routing for BGP

Border Gateway Protocol (BGP) is an Exterior Gateway Protocol (EGP) that allows you to create loop-free inter-domain routing between autonomous systems. An autonomous system is a set of routers under a single technical administration. Routers in an autonomous system can use multiple Interior Gateway Protocols (IGPs) to exchange routing information inside the autonomous system and an EGP to route packets outside the autonomous system.

This module provides the configuration information used to enable segment routing for BGP.

For additional information on implementing BGP on your, see the Implementing BGP module in the.

Note

- Segment Routing for BGP, on page 25
- Configure BGP Prefix Segment Identifiers, on page 26
- Configure Segment Routing Egress Peer Engineering, on page 27
- Configure BGP Link-State, on page 29
- Example: Configuring SR-EPE and BGP-LS, on page 30

Segment Routing for BGP

In a traditional BGP-based data center (DC) fabric, packets are forwarded hop-by-hop to each node in the autonomous system. Traffic is directed only along the external BGP (eBGP) multipath ECMP. No traffic engineering is possible.

In an MPLS-based DC fabric, the eBGP sessions between the nodes exchange BGP labeled unicast (BGP-LU) network layer reachability information (NLRI). An MPLS-based DC fabric allows any leaf (top-of-rack or border router) in the fabric to communicate with any other leaf using a single label, which results in higher packet forwarding performance and lower encapsulation overhead than traditional BGP-based DC fabric. However, since each label value might be different for each hop, an MPLS-based DC fabric is more difficult to troubleshoot and more complex to configure.

BGP has been extended to carry segment routing prefix-SID index. BGP-LU helps each node learn BGP prefix SIDs of other leaf nodes and can use ECMP between source and destination. Segment routing for BGP simplifies the configuration, operation, and troubleshooting of the fabric. With segment routing for BGP, you can enable traffic steering capabilities in the data center using a BGP prefix SID.
Configure BGP Prefix Segment Identifiers

Segments associated with a BGP prefix are known as BGP prefix SIDs. The BGP prefix SID is global within a segment routing or BGP domain. It identifies an instruction to forward the packet over the ECMP-aware best-path computed by BGP to the related prefix. The BGP prefix SID is manually configured from the segment routing global block (SRGB) range of labels.

Each BGP speaker must be configured with an SRGB using the `segment-routing global-block` command. See the About the Segment Routing Global Block section for information about the SRGB.

---

**Note**

Because the values assigned from the range have domain-wide significance, we recommend that all routers within the domain be configured with the same range of values.

---

To assign a BGP prefix SID, first create a routing policy using the `set label-index index` attribute, then associate the index to the node.

---

**Note**

A routing policy with the `set label-index` attribute can be attached to a network configuration or redistribute configuration. Other routing policy language (RPL) configurations are possible. For more information on routing policies, refer to the "Implementing Routing Policy" chapter in the Routing Configuration Guide for Cisco NCS 5000 Series Routers.

---

**Example**

The following example shows how to configure the SRGB, create a BGP route policy using a $SID parameter and `set label-index` attribute, and then associate the prefix-SID index to the node.

```
RP/0/RSP0/CPU0:router(config)# segment-routing global-block 16000 23999
RP/0/RSP0/CPU0:router(config)# route-policy SID($SID)
RP/0/RSP0/CPU0:router(config-rpl)# set label-index $SID
RP/0/RSP0/CPU0:router(config-rpl)# end policy
RP/0/RSP0/CPU0:router(config)# router bgp 1
RP/0/RSP0/CPU0:router(config-bgp)# bgp router-id 1.1.1.1
RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 unicast
RP/0/RSP0/CPU0:router(config-bgp-af)# network 1.1.1.3/32 route-policy SID(3)
RP/0/RSP0/CPU0:router(config-bgp-af)# allocate-label all
RP/0/RSP0/CPU0:router(config-bgp-af)# commit
RP/0/RSP0/CPU0:router(config)#
```

RP/0/RSP0/CPU0:router# show bgp 1.1.1.3/32
BGP routing table entry for 1.1.1.3/32
Versions:
  Process bRIB/RIB SendTblVer
  Speaker: 74 74
Local Label: 16003
Last Modified: Sep 29 19:52:18.155 for 00:07:22
Paths: (1 available, best #1)
  Advertised to update-groups (with more than one peer):
    0.2
Configure Segment Routing Egress Peer Engineering

Segment routing egress peer engineering (EPE) uses a controller to instruct an ingress provider edge, or a content source (node) within the segment routing domain, to use a specific egress provider edge (node) and a specific external interface to reach a destination. BGP peer SIDs are used to express source-routed inter-domain paths.

The controller learns the BGP peer SIDs and the external topology of the egress border router through BGP-LS EPE routes. The controller can program an ingress node to steer traffic to a destination through the egress node and peer node using BGP labeled unicast (BGP-LU).

EPE functionality is only required at the EPE egress border router and the EPE controller.

This task explains how to configure segment routing EPE on the EPE egress node.

**SUMMARY STEPS**

1. `router bgp as-number`
2. `neighbor ip-address`
3. `remote-as as-number`
4. `egress-engineering`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | `router bgp as-number` | Specifies the BGP AS number and enters the BGP configuration mode, allowing you to configure the BGP routing process.  
**Example:**
RP/0/RSP0/CPU0:router(config)# router bgp 1 |
| **Step 2** | `neighbor ip-address` | Places the router in neighbor configuration mode for BGP routing and configures the neighbor IP address as a BGP peer.  
**Example:**
RP/0/RSP0/CPU0:router(config-bgp)# neighbor 192.168.1.3 |
| **Step 3** | `remote-as as-number` | Creates a neighbor and assigns a remote autonomous system number to it.  
**Example:** |
### Understanding the ECMP Solution for BGP Labeled Unicast

This section explains the drawbacks of using the destination load balancer (DLB) algorithm for BGP labeled unicast (LU) and provides the premise for introducing the ECMP solution.

#### Drawbacks of Using the Destination Load Balancer Algorithm

A BGP-based data center fabric uses the DLB algorithm to choose the next-hop based on the destination prefix. As an example, consider a scenario with two provider edge routers: Router PE1 and Router PE2. Router PE1 has multiple paths to Router PE2, but only one path is chosen by the DLB algorithm as the best path. Traffic is sent only along the best path between routers PE1 and PE2, and the remaining paths are used for other destination prefixes.

Hence, if Router PE1 receives too much traffic destined for Router PE2, a single path is overloaded. The path overload leads to imbalance and congestion in the network.

#### ECMP for BGP LU

From Cisco IOS XR Release 6.3.1 onwards, routers using BGP LU can use ECMP to equally distribute the traffic along all available paths to a chosen destination. BGP uses the 5-tuple address hash for ECMP load balancing.

You can enable either the DLB or ECMP method of load balancing by using the `hw-module` command in global configuration mode.

---

**Note**

Cisco NCS 5000 Series Routers support the configuration of 8 BGP and 8 IGP paths with ECMP. However, even though the system supports the configuration of 64 (8*8) paths, only 32 paths can be processed at a time with ECMP.

---

### Enabling ECMP for BGP LU

This section explains how you can enable ECMP for BGP LU.

**Configuration**

Use the following configuration to enable ECMP for BGP LU.
Configure BGP Link-State

BGP Link-State (LS) is an Address Family Identifier (AFI) and Sub-address Family Identifier (SAFI) defined to carry interior gateway protocol (IGP) link-state database through BGP. BGP LS delivers network topology information to topology servers and Application Layer Traffic Optimization (ALTO) servers. BGP LS allows policy-based control to aggregation, information-hiding, and abstraction. BGP LS supports IS-IS and OSPFv2.

Note

IGPs do not use BGP LS data from remote peers. BGP does not download the received BGP LS data to any other component on the router.

For segment routing, the following attributes have been added to BGP LS:

- Node—Segment routing capability (including SRGB range) and algorithm
- Link—Adjacency SID and LAN adjacency SID
- Prefix—Prefix SID and segment routing mapping server (SRMS) prefix range

The following example shows how to exchange link-state information with a BGP neighbor:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# router bgp 1
RP/0/RSP0/CPU0:router(config-bgp)# neighbor 10.0.0.2
RP/0/RSP0/CPU0:router(config-bgp-nbr)# remote-as 1
RP/0/RSP0/CPU0:router(config-bgp-nbr)# address-family link-state link-state
RP/0/RSP0/CPU0:router(config-bgp-nbr-af)# exit
```

IGP Extensions

A given BGP node may have connections to multiple, independent routing domains; IGP link state distribution into BGP has been added for both OSPF and ISIS protocols to enable that node to pass this information, in a similar fashion, on to applications that desire to build paths spanning or including these multiple domains.

To distribute ISIS link-state data using BGP LS, use the `distribute bgp-ls` command in router configuration mode.

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# router isis lsp
```
Example: Configuring SR-EPE and BGP-LS

In the following figure, segment routing is enabled on autonomous system AS1 with ingress node A and egress nodes B and C. In this example, we configure EPE on egress node C.

**Figure 2: Topology**

Step 1  Configure node C with EPE for eBGP peers D and E.

**Example:**

```markdown
RP/0/RSP0/CPU0:router_C(config)# router bgp 1
RP/0/RSP0/CPU0:router_C(config-bgp)# neighbor 192.168.1.3
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# remote-as 3
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# description to E
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# egress-engineering
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# address-family ipv4 unicast
RP/0/RSP0/CPU0:router_C(config-bgp-nbr-af)# route-policy bgp_in in
RP/0/RSP0/CPU0:router_C(config-bgp-nbr-af)# route-policy bgp_out out
RP/0/RSP0/CPU0:router_C(config-bgp-nbr-af)# exit
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# exit
RP/0/RSP0/CPU0:router_C(config-bgp)# neighbor 192.168.1.2
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# remote-as 2
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# description to D
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# egress-engineering
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# address-family ipv4 unicast
RP/0/RSP0/CPU0:router_C(config-bgp-nbr-af)# route-policy bgp_in in
RP/0/RSP0/CPU0:router_C(config-bgp-nbr-af)# route-policy bgp_out out
RP/0/RSP0/CPU0:router_C(config-bgp-nbr-af)# exit
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# exit
```

Step 2  Configure node C to advertise peer node SIDs to the controller using BGP-LS.
Example:

RP/0/RSP0/CPU0:router_C(config-bgp)# neighbor 172.29.50.71
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# remote-as 1
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# description to EPE_controller
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# address-family link-state link-state
RP/0/RSP0/CPU0:router_C(config-bgp-nbr)# exit
RP/0/RSP0/CPU0:router_C(config-bgp)# exit

Step 3  Commit the configuration.

Example:

RP/0/RSP0/CPU0:router_C(config)# commit

Step 4  Verify the configuration.

Example:

RP/0/RSP0/CPU0:router_C# show bgp egress-engineering

Egress Engineering Peer Set: 192.168.1.2/32 (10b87210)
  Nexthop: 192.168.1.2
  Version: 2, rn_version: 2
  Flags: 0x00000002
  Local ASN: 1
  Remote ASN: 2
  Local RID: 1.1.1.3
  Remote RID: 1.1.1.4
  First Hop: 192.168.1.2
  NHID: 3
  Label: 24002, Refcount: 3
  rpc_set: 10b9d408

Egress Engineering Peer Set: 192.168.1.3/32 (10be61d4)
  Nexthop: 192.168.1.3
  Version: 3, rn_version: 3
  Flags: 0x00000002
  Local ASN: 1
  Remote ASN: 3
  Local RID: 1.1.1.3
  Remote RID: 1.1.1.5
  First Hop: 192.168.1.3
  NHID: 4
  Label: 24003, Refcount: 3
  rpc_set: 10be6250

The output shows that node C has allocated peer SIDs for each eBGP peer.

Example:

RP/0/RSP0/CPU0:router_C# show mpls forwarding labels 24002 24003

<table>
<thead>
<tr>
<th>Label</th>
<th>Prefix</th>
<th>Outgoing</th>
<th>Next Hop</th>
<th>Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>24002</td>
<td>Unlabelled</td>
<td>Te0/3/0/0</td>
<td>192.168.1.2</td>
<td>0</td>
</tr>
<tr>
<td>24003</td>
<td>Unlabelled</td>
<td>Te0/1/0/0</td>
<td>192.168.1.3</td>
<td>0</td>
</tr>
</tbody>
</table>
The output shows that node C installed peer node SIDs in the Forwarding Information Base (FIB).
Configure Topology-Independent Loop-Free Alternate (TI-LFA)

Topology-Independent Loop-Free Alternate (TI-LFA) uses segment routing to provide link protection in topologies where other fast reroute techniques cannot provide protection. The goal of TI-LFA is to reduce the packet loss that results while routers converge after a topology change due to a link failure. Rapid failure repair (< 50 msec) is achieved through the use of pre-calculated backup paths that are loop-free and safe to use until the distributed network convergence process is completed.

Note
TI-LFA supports IPv4 only.

TI-LFA provides link protection. The link is excluded during the post convergence backup path calculation.

- Configuring TI-LFA for IS-IS, on page 33
- Configuring TI-LFA for OSPF, on page 35

Configuring TI-LFA for IS-IS

This task describes how to enable per-prefix Topology Independent Loop-Free Alternate (TI-LFA) computation to converge traffic flows around link failures.

Before you begin

Ensure that the following topology requirements are met:

- Router interfaces are configured as per the topology.
- Routers are configured with IS-IS.
- Segment routing for IS-IS is configured. See Enabling Segment Routing for IS-IS Protocol, on page 11.
- Enter the following commands in global configuration mode:

  ```bash
  Router(config)# ipv4 unnumbered mpls traffic-eng Loopback0
  Router(config)# mpls traffic-eng
  Router(config-mpls-te)# exit
  Router(config)#
  ```
SUMMARY STEPS

1. configure
2. router isis instance-id
3. interface type interface-path-id
4. address-family ipv4 [unicast]
5. fast-reroute per-prefix
6. fast-reroute per-prefix ti-lfa

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td>Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong> router isis instance-id</td>
<td>You can change the level of routing to be performed by a particular routing instance by using the <code>is-type</code> router configuration command.</td>
</tr>
<tr>
<td>Example: <code>RP/0/RP0/CPU0:router(config)# router isis 1</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> interface type interface-path-id</td>
<td>Enters interface configuration mode.</td>
</tr>
<tr>
<td>Example: <code>RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet0/0/2/1</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> address-family ipv4 [unicast]</td>
<td>Specifies the IPv4 address family, and enters router address family configuration mode.</td>
</tr>
<tr>
<td>Example: <code>RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> fast-reroute per-prefix</td>
<td>Enables per-prefix fast reroute.</td>
</tr>
<tr>
<td>Example: <code>RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> fast-reroute per-prefix ti-lfa</td>
<td>Enables per-prefix TI-LFA fast reroute link protection.</td>
</tr>
<tr>
<td>Example: <code>RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix ti-lfa</code></td>
<td></td>
</tr>
</tbody>
</table>

TI-LFA has been successfully configured for segment routing.
Configuring TI-LFA for OSPF

This task describes how to enable per-prefix Topology Independent Loop-Free Alternate (TI-LFA) computation to converge traffic flows around link failures.

**Note**

TI-LFA can be configured on the instance, area, or interface. When configured on the instance or area, all interfaces in the instance or area inherit the configuration.

**Before you begin**

Ensure that the following topology requirements are met:

- Router interfaces are configured as per the topology.
- Routers are configured with OSPF.
- Segment routing for OSPF is configured. See Enabling Segment Routing for OSPF Protocol, on page 21.
- Enter the following commands in global configuration mode:

  ```
  Router(config)# ipv4 unnumbered mpls traffic-eng Loopback0
  Router(config)# mpls traffic-eng
  Router(config-mpls-te)# exit
  Router(config)#
  ```

**SUMMARY STEPS**

1. `configure`
2. `router ospf process-name`
3. `area area-id`
4. `interface type interface-path-id`
5. `fast-reroute per-prefix`
6. `fast-reroute per-prefix ti-lfa`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure</td>
<td>Enables OSPF routing for the specified routing process, and places the router in router configuration mode.</td>
</tr>
<tr>
<td>Step 2</td>
<td><code>router ospf process-name</code> Example: <code>RP/0/RP0/CPU0:router(config)# router ospf 1</code></td>
<td>Enters area configuration mode.</td>
</tr>
<tr>
<td>Step 3</td>
<td><code>area area-id</code> Example:</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RP0/CPU0:router(config-ospf)# area 1</code></td>
<td>Enters interface configuration mode.</td>
<td></td>
</tr>
</tbody>
</table>
| **Step 4** interface type interface-path-id  
Example: `RP/0/RP0/CPU0:router(config-ospf-ar)# interface GigabitEthernet0/0/2/1` | Enables per-prefix fast reroute. |
| **Step 5** fast-reroute per-prefix  
Example: `RP/0/RP0/CPU0:router(config-ospf-ar-if)# fast-reroute per-prefix` | Enables per-prefix TI-LFA fast reroute link protection. |
| **Step 6** fast-reroute per-prefix ti-lfa  
Example: `RP/0/RP0/CPU0:router(config-ospf-ar-if)# fast-reroute per-prefix ti-lfa` | |

TI-LFA has been successfully configured for segment routing.
Configure Segment Routing Mapping Server

The mapping server is a key component of the interworking between LDP and segment routing. It enables SR-capable nodes to interwork with LDP nodes. The mapping server advertises Prefix-to-SID mappings in IGP on behalf of other non-SR-capable nodes.

- Segment Routing Mapping Server, on page 37
- Segment Routing and LDP Interoperability, on page 38
- Configuring Mapping Server, on page 40
- Enable Mapping Advertisement, on page 42
- Enable Mapping Client, on page 44

Segment Routing Mapping Server

The mapping server functionality in Cisco IOS XR segment routing centrally assigns prefix-SIDs for some or all of the known prefixes. A router must be able to act as a mapping server, a mapping client, or both.

- A router that acts as a mapping server allows the user to configure SID mapping entries to specify the prefix-SIDs for some or all prefixes. This creates the local SID-mapping policy. The local SID-mapping policy contains non-overlapping SID-mapping entries. The mapping server advertises the local SID-mapping policy to the mapping clients.

- A router that acts as a mapping client receives and parses remotely received SIDs from the mapping server to create remote SID-mapping entries.

- A router that acts as a mapping server and mapping client uses the remotely learnt and locally configured mapping entries to construct the non-overlapping consistent active mapping policy. IGP instance uses the active mapping policy to calculate the prefix-SIDs of some or all prefixes.

The mapping server automatically manages the insertions and deletions of mapping entries to always yield an active mapping policy that contains non-overlapping consistent SID-mapping entries.

- Locally configured mapping entries must not overlap each other.

- The mapping server takes the locally configured mapping policy, as well as remotely learned mapping entries from a particular IGP instance, as input, and selects a single mapping entry among overlapping mapping entries according to the preference rules for that IGP instance. The result is an active mapping policy that consists of non-overlapping consistent mapping entries.

- At steady state, all routers, at least in the same area or level, must have identical active mapping policies.
Segment Routing Mapping Server Restrictions

- The position of the mapping server in the network is not important. However, since the mapping advertisements are distributed in IGP using the regular IGP advertisement mechanism, the mapping server needs an IGP adjacency to the network.

- The role of the mapping server is crucial. For redundancy purposes, you should configure multiple mapping servers in the networks.

- The mapping server functionality does not support a scenario where SID-mapping entries learned through one IS-IS instance are used by another IS-IS instance to determine the prefix-SID of a prefix. For example, mapping entries learnt from remote routers by 'router isis 1' cannot be used to calculate prefix-SIDs for prefixes learnt, advertised, or downloaded to FIB by 'router isis 2'. A mapping server is required for each IS-IS area.

- Segment Routing Mapping Server does not support Virtual Routing and Forwarding (VRF) currently.

Segment Routing and LDP Interoperability

IGP provides mechanisms through which segment routing (SR) interoperate with label distribution protocol (LDP). The control plane of segment routing co-exists with LDP.

The Segment Routing Mapping Server (SRMS) functionality in SR is used to advertise SIDs for destinations, in the LDP part of the network, that do not support SR. SRMS maintains and advertises segment identifier (SID) mapping entries for such destinations. IGP propagates the SRMS mapping entries and interacts with SRMS to determine the SID value when programming the forwarding plane. IGP installs prefixes and corresponding labels, into routing information base (RIB), that are used to program the forwarding information base (FIB).

Example: Segment Routing LDP Interoperability

Consider a network with a mix of segment routing (SR) and label distribution protocol (LDP). A continuous multiprotocol label switching (MPLS) LSP (Labeled Switched Path) can be established by facilitating interoperability. One or more nodes in the SR domain act as segment routing mapping server (SRMS). SRMS advertises SID mappings on behalf of non-SR capable nodes. Each SR-capable node learns about SID assigned to non-SR capable nodes without explicitly configuring individual nodes.

Consider a network as shown in the following image. This network is a mix of both LDP and SR-capable nodes.

In this mixed network:

- Nodes P6, P7, P8, PE4 and PE3 are LDP-capable
- Nodes PE1, PE2, P5 and P6 are SR-capable
- Nodes PE1, PE2, P5 and P6 are configured with segment routing global block (SRGB) of (100, 200)
- Nodes PE1, PE2, P5 and P6 are configured with node segments of 101, 102, 105 and 106 respectively
A service flow must be established from PE1 to PE3 over a continuous MPLS tunnel. This requires SR and LDP to interoperate.

**LDP to SR**

The traffic flow from LDP to SR (right to left) involves:

1. PE3 learns a service route whose nhop is PE1. PE3 has an LDP label binding from the nhop P8 for the FEC PE1. PE3 forwards the packet P8.
2. P8 has an LDP label binding from its nhop P7 for the FEC PE1. P8 forwards the packet to P7.
3. P7 has an LDP label binding from its nhop P6 for the FEC PE1. P7 forwards the packet to P6.
4. P6 does not have an LDP binding from its nhop P5 for the FEC PE1. But P6 has an SR node segment to the IGP route PE1. P6 forwards the packet to P5 and swaps its local LDP label for FEC PE1 by the equivalent node segment 101. This process is called label merging.
5. P5 pops 101, assuming PE1 has advertised its node segment 101 with the penultimate-pop flag set and forwards to PE1.
6. PE1 receives the tunneled packet and processes the service label.

The end-to-end MPLS tunnel is established from an LDP LSP from PE3 to P6 and the related node segment from P6 to PE1.

**SR to LDP**

Suppose that the operator configures P5 as a Segment Routing Mapping Server (SRMS) and advertises the mappings (P7, 107), (P8, 108), (PE3, 103) and (PE4, 104). If PE3 was SR-capable, the operator may have configured PE3 with node segment 103. Because PE3 is non-SR capable, the operator configures that policy at the SRMS; the SRMS advertises the mapping on behalf of the non-SR capable nodes. Multiple SRMS servers can be provisioned in a network for redundancy. The mapping server advertisements are only understood by the SR-capable nodes. The SR capable routers install the related node segments in the MPLS data plane in exactly the same manner if node segments were advertised by the nodes themselves.

The traffic flow from SR to LDP (left to right) involves:

1. PE1 installs the node segment 103 with nhop P5 in exactly the same manner if PE3 had advertised node segment 103.
2. P5 swaps 103 for 103 and forwards to P6.
3. The nhop for P6 for the IGP route PE3 is non-SR capable. (P7 does not advertise the SR capability.) However, P6 has an LDP label binding from that nhop for the same FEC. (For example, LDP label 1037.) P6 swaps 103 for 1037 and forwards to P7. We refer to this process as label merging.
4. P7 swaps this label with the LDP label received from P8 and forwards to P8.
5. P8 pops the LDP label and forwards to PE3.
6. PE3 receives the packet and processes as required.

The end-to-end MPLS LSP is established from an SR node segment from PE1 to P6 and an LDP LSP from P6 to PE3.
Configuring Mapping Server

Perform these tasks to configure the mapping server and to add prefix-SID mapping entries in the active local mapping policy.

SUMMARY STEPS

1. configure
2. segment-routing
3. mapping-server
4. prefix-sid-map
5. address-family ipv4 | ipv6
6. ip-address / prefix-length first-SID-value range range
7. commit
8. show segment-routing mapping-server prefix-sid-map [ipv4 | ipv6] [detail]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>configure</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>segment-routing</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RP0/CPU0:router(config)# segment-routing</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>mapping-server</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RP0/CPU0:router(config-sr)# mapping-server</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>prefix-sid-map</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RP0/CPU0:router(config-sr-ms)# prefix-sid-map</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>address-family ipv4</td>
</tr>
<tr>
<td>Example:</td>
<td>This example shows the address-family for ipv4:</td>
</tr>
<tr>
<td></td>
<td>This example shows the address-family for ipv6:</td>
</tr>
</tbody>
</table>
### Configure Segment Routing Mapping Server

**Configure Segment Routing Mapping Server**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| RP/0/RP0/CPU0:router(config-sr-ms-map)# address-family ipv6 | Adds SID-mapping entries in the active local mapping policy. In the configured example:  
- Prefix 10.1.1.1/32 is assigned prefix-SID 10, prefix 10.1.1.2/32 is assigned prefix-SID 11,..., prefix 10.1.1.199/32 is assigned prefix-SID 200  
- Prefix 20.1.0.0/16 is assigned prefix-SID 400, prefix 20.2.0.0/16 is assigned prefix-SID 401,..., and so on. |

**Step 6**  
*ip-address/prefix-length first-SID-value range range*  
**Example:**  
RP/0/RP0/CPU0:router(config-sr-ms-map-af)#  
10.1.1.1/32 10 range 200  
RP/0/RP0/CPU0:router(config-sr-ms-map-af)#  
20.1.0.0/16 400 range 300  

**Step 7**  
*commit*  

**Step 8**  
*show segment-routing mapping-server prefix-sid-map [ipv4 | ipv6] [detail]*  
**Example:**  
RP/0/RP0/CPU0:router# show segment-routing  
mapping-server prefix-sid-map ipv4  
Prefix | SID Index | Range  
--- | --- | ---  
20.1.1.0/24 | 400 | 300  
10.1.1.1/32 | 10 | 200  

Number of mapping entries: 2  
RP/0/RP0/CPU0:router# show segment-routing  
mapping-server prefix-sid-map ipv4 detail  
Prefix  
20.1.1.0/24  
| SID Index: | 400  
| Range: | 300  
| Last Prefix: | 20.2.44.0/24  
| Last SID Index: | 699  
| Flags: |  
10.1.1.1/32  
| SID Index: | 10  
| Range: | 200  
| Last Prefix: | 10.1.1.200/32  
| Last SID Index: | 209  
| Flags: |  

Number of mapping entries: 2  

**What to do next**  
Enable the advertisement of the local SID-mapping policy in the IGP.
Enable Mapping Advertisement

In addition to configuring the static mapping policy, you must enable the advertisement of the mappings in the IGP.

Perform these steps to enable the IGP to advertise the locally configured prefix-SID mapping.

Configure Mapping Advertisement for IS-IS

SUMMARY STEPS

1. router isis instance-id
2. address-family { ipv4 | ipv6 } [ unicast ]
3. segment-routing prefix-sid-map advertise-local
4. commit
5. show isis database verbose

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.</td>
</tr>
<tr>
<td>router isis instance-id</td>
<td>Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.</td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config)# router isis 1</td>
<td>Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.</td>
</tr>
<tr>
<td>address-family { ipv4</td>
<td>ipv6 } [ unicast ]</td>
</tr>
<tr>
<td>Example:</td>
<td>Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.</td>
</tr>
<tr>
<td>The following is an example for ipv4 address family:</td>
<td>Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.</td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast</td>
<td>Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Configures IS-IS to advertise locally configured prefix-SID mappings.</td>
</tr>
<tr>
<td>segment-routing prefix-sid-map advertise-local</td>
<td>Configures IS-IS to advertise locally configured prefix-SID mappings.</td>
</tr>
<tr>
<td>Example:</td>
<td>Configures IS-IS to advertise locally configured prefix-SID mappings.</td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config-isis-af)# segment-routing prefix-sid-map advertise-local</td>
<td>Configures IS-IS to advertise locally configured prefix-SID mappings.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>commit</td>
<td>Configures IS-IS to advertise locally configured prefix-SID mappings.</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>Displays IS-IS prefix-SID mapping advertisement and TLV.</td>
</tr>
<tr>
<td>show isis database verbose</td>
<td>Displays IS-IS prefix-SID mapping advertisement and TLV.</td>
</tr>
<tr>
<td>Example:</td>
<td>Displays IS-IS prefix-SID mapping advertisement and TLV.</td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router# show isis database verbose</td>
<td>Displays IS-IS prefix-SID mapping advertisement and TLV.</td>
</tr>
</tbody>
</table>
Configure Mapping Advertisement for OSPF

SUMMARY STEPS

1. `router ospf process-name`
2. `segment-routing prefix-sid-map advertise-local`
3. `commit`
4. `show ospf database opaque-area`

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 1 | `router ospf process-name`  
Example:  
RP/0/RP0/CPU0:router(config)# `router ospf 1` | Enables OSPF routing for the specified routing instance, and places the router in router configuration mode. |
| Step 2 | `segment-routing prefix-sid-map advertise-local`  
Example:  
RP/0/RP0/CPU0:router(config-ospf)# `segment-routing prefix-sid-map advertise-local` | Configures OSPF to advertise locally configured prefix-SID mappings. |
| Step 3 | `commit` | |
| Step 4 | `show ospf database opaque-area`  
Example:  
RP/0/RP0/CPU0:router# `show ospf database opaque-area`  
<...removed...>  
Extended Prefix Range TLV: Length: 24  
AF: 0  
Prefix: 10.1.1.1/32  
Range Size: 200  
Flags: 0x0  
SID sub-TLV: Length: 8 | Displays OSPF prefix-SID mapping advertisement and TLV. |
Enable Mapping Client

By default, mapping client functionality is enabled.

You can disable the mapping client functionality by using the `segment-routing prefix-sid-map receive disable` command.

You can re-enable the mapping client functionality by using the `segment-routing prefix-sid-map receive` command.

The following example shows how to enable the mapping client for IS-IS:

```
RP/0/RP0/CPU0:router(config)# router isis 1
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# segment-routing prefix-sid-map receive
```

The following example shows how to enable the mapping client for OSPF:

```
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# segment-routing prefix-sid-map receive
```
Using Segment Routing OAM

Segment Routing Operations, Administration, and Maintenance (OAM) helps service providers to monitor label-switched paths (LSPs) and quickly isolate forwarding problems to assist with fault detection and troubleshooting in the network. The Segment Routing OAM feature provides support for Nil-FEC (forwarding equivalence classes) LSP Ping and Traceroute functionality.

- MPLS Ping and Traceroute for BGP and IGP Prefix-SID, on page 45
- Examples: MPLS Ping, Traceroute, and Tree Trace for Prefix-SID, on page 46
- MPLS LSP Ping and Traceroute Nil FEC Target, on page 47
- Examples: LSP Ping and Traceroute for Nil_FEC Target, on page 48

MPLS Ping and Traceroute for BGP and IGP Prefix-SID

MPLS Ping and Traceroute operations for Prefix SID are supported for various IGP scenarios, for example:

- Within an IS-IS level or OSPF area
- Across IS-IS levels or OSPF areas
- Route redistribution from IS-IS to OSPF and from OSPF to IS-IS
- Anycast Prefix SID

The MPLS LSP Ping feature is used to check the connectivity between ingress Label Switch Routers (LSRs) and egress LSRs along an LSP. MPLS LSP ping uses MPLS echo request and reply messages, similar to Internet Control Message Protocol (ICMP) echo request and reply messages, to validate an LSP. The destination IP address of the MPLS echo request packet is different from the address used to select the label stack. The destination IP address is defined as a 127.x.y.z/8 address and it prevents the IP packet from being IP switched to its destination, if the LSP is broken.

The MPLS LSP Traceroute feature is used to isolate the failure point of an LSP. It is used for hop-by-hop fault localization and path tracing. The MPLS LSP Traceroute feature relies on the expiration of the Time to Live (TTL) value of the packet that carries the echo request. When the MPLS echo request message hits a transit node, it checks the TTL value and if it is expired, the packet is passed to the control plane, else the message is forwarded. If the echo message is passed to the control plane, a reply message is generated based on the contents of the request message.

The MPLS LSP Tree Trace (traceroute multipath) operation is also supported for IGP Prefix SID. MPLS LSP Tree Trace provides the means to discover all possible equal-cost multipath (ECMP) routing paths of an LSP to reach a destination Prefix SID. It uses multipath data encoded in echo request packets to query for the
Examples: MPLS Ping, Traceroute, and Tree Trace for Prefix-SID

These examples use the following topology:

```
R1 1.1.1.1
  F2
  |
  R3
  |
R4 1.1.1.4
```

### MPLS Ping for Prefix-SID

RP/0/RP0/CPU0:router-arizona# ping mpls ipv4 1.1.1.4/32
Thu Dec 17 01:01:42.301 PST

Sending 5, 100-byte MPLS Echos to 1.1.1.4, timeout is 2 seconds, send interval is 0 msecs:

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

Type escape sequence to abort.

!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms

### MPLS Traceroute for Prefix-SID

RP/0/RP0/CPU0:router-arizona# traceroute mpls ipv4 1.1.1.4/32
Thu Dec 17 14:45:05.563 PST

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'I' - FEC mismatch,
'M' - malformed request, 'm' - unsupported tlv, 'N' - no rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,

Using Segment Routing OAM
'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

0 12.12.12.1 MRU 4470 [Labels: 16004 Exp: 0]
L 1 12.12.12.2 MRU 4470 [Labels: 16004 Exp: 0] 3 ms
L 2 23.23.23.3 MRU 4470 [Labels: implicit-null Exp: 0] 3 ms
! 3 34.34.34.4 11 ms

**MPLS Tree Trace for Prefix-SID**

RP/0/RP0/CPU0:router-arizona# traceroute mpls multipath ipv4 1.1.1.4/32
Thu Dec 17 14:55:46.549 PST

Starting LSP Path Discovery for 1.1.1.4/32

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 rx label,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.

LL!
Path 0 found,
  output interface TenGigE0/0/0/0 nexthop 12.12.12.2 source 12.12.12.1 destination 127.0.0.0
L!
Path 1 found,
  output interface TenGigE0/0/0/0 nexthop 12.12.12.2 source 12.12.12.1 destination 127.0.0.2
LL!
Path 2 found,
  output interface TenGigE0/0/0/1 nexthop 15.15.15.5 source 15.15.15.1 destination 127.0.0.1
L!
Path 3 found,
  output interface TenGigE0/0/0/1 nexthop 15.15.15.5 source 15.15.15.1 destination 127.0.0.0

Paths (found/broken/unexplored) (4/0/0)
Echo Request (sent/fail) (10/0)
Echo Reply (received/timeout) (10/0)
Total Time Elapsed 53 ms

**MPLS LSP Ping and Traceroute Nil FEC Target**

The Nil-FEC LSP ping and traceroute operations are extensions of regular MPLS ping and traceroute.

Nil-FEC LSP Ping/Traceroute functionality supports segment routing and MPLS Static. It also acts 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 adjacency label along the routing path is put into the label stack of an echo request message from the initiator Label Switch Router (LSR); MPLS data plane forwards this packet to the label stack target, and the label stack target sends the echo message back.

The following table shows the syntax for the ping and traceroute commands.

### Table 2: LSP Ping and Traceroute Nil FEC Commands

<table>
<thead>
<tr>
<th>Command Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>ping mpls nil-fec labels {label[, label]} {output {interface tx-interface} {nexthop nexthop-ip-addr}}</td>
</tr>
<tr>
<td>traceroute mpls nil-fec labels {label[, label]} {output {interface tx-interface} {nexthop nexthop-ip-addr}}</td>
</tr>
</tbody>
</table>

### Examples: LSP Ping and Traceroute for Nil_FEC Target

These examples use the following topology:

- **Node loopback IP address:** 172.18.1.3 172.18.1.4 172.18.1.5 172.18.1.7
- **Node label:** 16004 16005 16007
- **Nodes:** Arizona ---- Utah ------- Wyoming ---- Texas
- **Interface:** GigabitEthernet0/2/0/1 GigabitEthernet0/2/0/1
- **Interface IP address:** 10.1.1.3 10.1.1.4

```
RP/0/RP0/CPU0# show mpls forwarding
```

```
Tue Jul  5 13:44:31.999 EDT
Local Label Label or ID Outgoing Prefix Outgoing Next Hop Bytes Switched
---------- ---------- ------------------ ------------------ ------------- --------------- ---------
16004 Pop No ID Gi0/2/0/1 10.1.1.4 1392
   Pop No ID Gi0/2/0/2 10.1.2.2 0
16005 16005 No ID Gi0/2/0/0 10.1.1.4 0
   16005 No ID Gi0/2/0/1 10.1.2.2 0
16007 16007 No ID Gi0/2/0/0 10.1.1.4 4752
   16007 No ID Gi0/2/0/1 10.1.2.2 0
24000 Pop SR Adj (idx 0) Gi0/2/0/0 10.1.1.4 0
24001 Pop SR Adj (idx 2) Gi0/2/0/0 10.1.1.4 0
24002 Pop SR Adj (idx 0) Gi0/2/0/1 10.1.2.2 0
24003 Pop SR Adj (idx 2) Gi0/2/0/1 10.1.2.2 0
24004 Pop No ID tt10 point2point 0
24005 Pop No ID tt11 point2point 0
24006 Pop No ID tt12 point2point 0
24007 Pop No ID tt13 point2point 0
24008 Pop No ID tt30 point2point 0
```

**Ping Nil FEC Target**

```
RP/0/RP0/CPU0# ping mpls nil-fec labels 16005,16007 output interface GigabitEthernet 0/2/0/0/1 nexthop 10.1.1.4 repeat 1
Sending 1, 72-byte MPLS Echos with Nil FEC labels 16005,16007, timeout is 2 seconds, send interval is 0.05 s:
```
Success rate is 100 percent (1/1), round-trip min/avg/max = 1/1/1 ms
Total Time Elapsed 0 ms

Traceroute Nil FEC Target

RP/0/RP0/CPU0:router-arizona# traceroute mpls nil-fec labels 16005,16007 output interface GigabitEthernet 0/2/0/1 nexthop 10.1.1.4
Tracing MPLS Label Switched Path with Nil FEC labels 16005,16007, timeout is 2 seconds


Type escape sequence to abort.

0 10.1.1.3 MRU 1500 [Labels: 16005/16007/explicit-null Exp: 0/0/0]
L 1 10.1.1.4 MRU 1500 [Labels: implicit-null/16007/explicit-null Exp: 0/0/0] 1 ms
L 2 10.1.1.5 MRU 1500 [Labels: implicit-null/explicit-null Exp: 0/0] 1 ms
! 3 10.1.1.7 1 ms
Examples: LSP Ping and Traceroute for Nil_FEC Target