



## **Segment Routing Configuration Guide for Cisco NCS 5000 Series Routers, IOS XR Release 7.9.x**

**First Published:** 2023-03-31

**Last Modified:** 2023-05-05

### **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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at [www.cisco.com/go/offices](http://www.cisco.com/go/offices).

The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on standards documentation, or language that is used by a referenced third-party product.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2023 Cisco Systems, Inc. All rights reserved.



## 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

#### **YANG Data Models for Segment Routing Features** 3

Using YANG Data Models 3

---

### CHAPTER 3

#### **About Segment Routing** 5

Scope 5

Need 6

Benefits 6

Workflow for Deploying Segment Routing 6

---

### CHAPTER 4

#### **Configure Segment Routing Global Block** 9

About the Segment Routing Global Block 9

Setup a Non-Default Segment Routing Global Block Range 11

---

### CHAPTER 5

#### **Configure Segment Routing for IS-IS Protocol** 13

Enabling Segment Routing for IS-IS Protocol 13

Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface 15

IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability 17

Prefix Attribute Flags 17

IPv4 and IPv6 Source Router ID 18

Configuring Prefix Attribute N-flag-clear	19
IS-IS Multi-Domain Prefix SID and Domain Stitching: Example	20
Configure IS-IS Multi-Domain Prefix SID	21
Configure Common Router ID	22
Distribute IS-IS Link-State Data	22

---

<b>CHAPTER 6</b>	<b>Configure Segment Routing for OSPF Protocol</b>	<b>25</b>
	Enabling Segment Routing for OSPF Protocol	25
	Configuring a Prefix-SID on the OSPF-Enabled Loopback Interface	27
	Configuring an Adjacency SID	29

---

<b>CHAPTER 7</b>	<b>Configure Segment Routing for BGP</b>	<b>33</b>
	Segment Routing for BGP	33
	Configure BGP Prefix Segment Identifiers	34
	Segment Routing Egress Peer Engineering	35
	Configure Segment Routing Egress Peer Engineering	35
	Understanding the ECMP Solution for BGP Labeled Unicast	37
	Enabling ECMP for BGP LU	37
	Configure BGP Link-State	38
	Use Case: Configuring SR-EPE and BGP-LS	40

---

<b>CHAPTER 8</b>	<b>Configure Topology-Independent Loop-Free Alternate (TI-LFA)</b>	<b>43</b>
	Usage Guidelines and Limitations	44
	Configuring TI-LFA for IS-IS	45
	Configuring TI-LFA for OSPF	47

---

<b>CHAPTER 9</b>	<b>Configure Segment Routing Mapping Server</b>	<b>49</b>
	Segment Routing Mapping Server	49
	Usage Guidelines and Restrictions	50
	Segment Routing and LDP Interoperability	50
	Example: Segment Routing LDP Interoperability	50
	Configuring Mapping Server	53
	Enable Mapping Advertisement	55
	Configure Mapping Advertisement for IS-IS	55

Configure Mapping Advertisement for OSPF	56
Enable Mapping Client	57

---

**CHAPTER 10****Using Segment Routing OAM 59**

MPLS Ping and Traceroute for BGP and IGP Prefix-SID	59
Examples: MPLS Ping, Traceroute, and Tree Trace for Prefix-SID	60
MPLS LSP Ping and Traceroute Nil FEC Target	61
Examples: LSP Ping and Traceroute for Nil_FEC Target	62





## 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.

Date	Change Summary
March 2023	Initial release of this document

## 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.







# CHAPTER 1

## 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

### Segment Routing Features Added or Modified in IOS XR Release 7.9.x

Feature	Description	Introduced/Changed in Release	Where Documented
None	None	None	None





## CHAPTER 2

# YANG Data Models for Segment Routing Features

This chapter provides information about the YANG data models for Segment Routing features.

- [Using YANG Data Models, on page 3](#)

## Using YANG Data Models

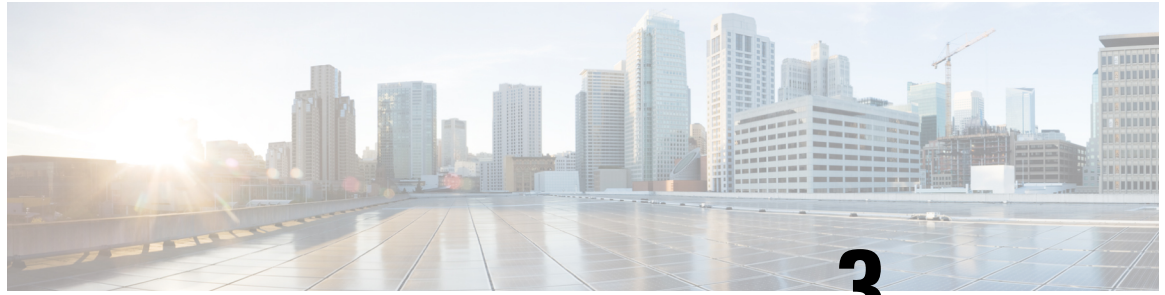
Cisco IOS XR supports a programmatic way of configuring and collecting operational data of a network device using YANG data models. Although configurations using CLIs are easier and human-readable, automating the configuration using model-driven programmability results in scalability.

The data models are available in the release image, and are also published in the [Github](#) repository. Navigate to the release folder of interest to view the list of supported data models and their definitions. Each data model defines a complete and cohesive model, or augments an existing data model with additional XPath. To view a comprehensive list of the data models supported in a release, navigate to the **Available-Content.md** file in the repository. The unsupported sensor paths are documented as deviations. For example, `openconfig-acl.yang` provides details about the supported sensor paths, whereas `cisco-xr-openconfig-acl-deviations.yang` shows the unsupported sensor paths for `openconfig-acl.yang` model on Cisco IOS XR routers.

You can also view the data model definitions using the [YANG Data Models Navigator](#) tool. This GUI-based and easy-to-use tool helps you explore the nuances of the data model and view the dependencies between various containers in the model. You can view the list of models supported across Cisco IOS XR releases and platforms, locate a specific model, view the containers and their respective lists, leaves, and leaf lists presented visually in a tree structure. This visual tree form helps you get insights into nodes that can help you automate your network.

To get started with using the data models, see the *Programmability Configuration Guide for Cisco NCS 5000 Series Routers*.





## CHAPTER 3

# About Segment Routing

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

- [Scope, on page 5](#)
- [Need, on page 6](#)
- [Benefits, on page 6](#)
- [Workflow for Deploying Segment Routing, on page 6](#)

## 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 for the IGP

3. Configure Segment Routing for BGP
4. Configure the Segment Routing Mapping Server







# CHAPTER 4

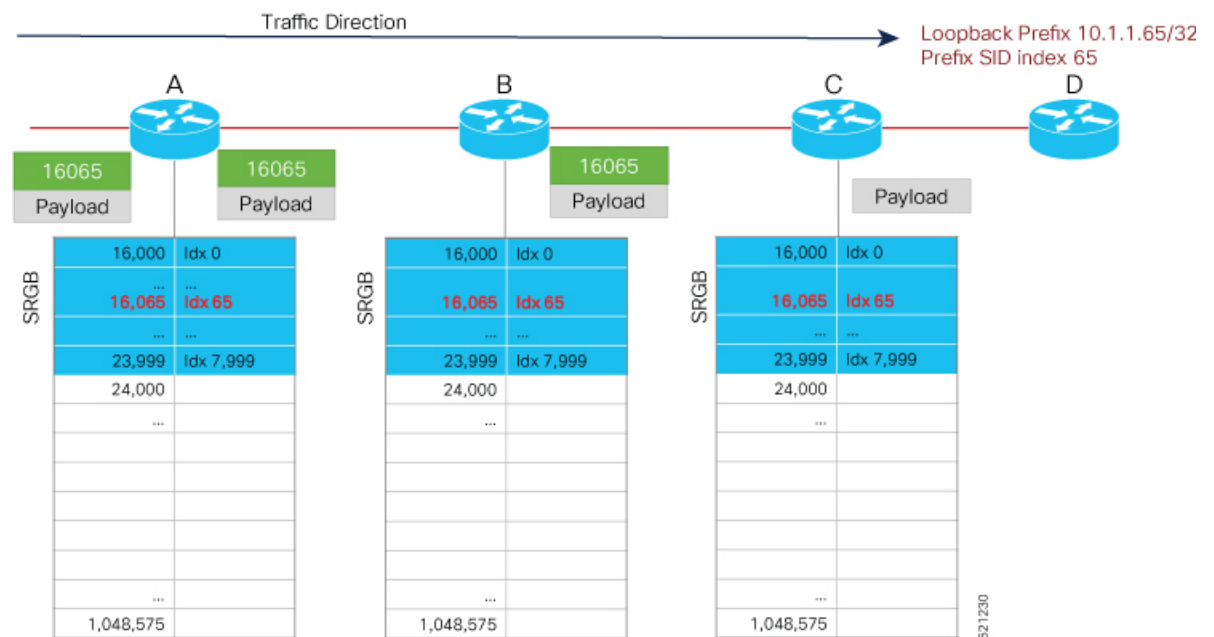
## Configure Segment Routing Global Block

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 9](#)
- [Setup a Non-Default Segment Routing Global Block Range, on page 11](#)

### About the Segment Routing Global Block

The Segment Routing Global Block (SRGB) is a range of labels reserved for Segment Routing global segments. A prefix-SID is advertised as a domain-wide unique index. The prefix-SID index points to a unique label within the SRGB range. The index is zero-based, meaning that the first index is 0. The MPLS label assigned to a prefix is derived from the Prefix-SID index plus the SRGB base. For example, considering an SRGB range of 16,000 to 23,999, a prefix 10.1.1.65/32 with prefix-SID index of **65** is assigned the label value of **16065**.



To keep the configuration simple and straightforward, we strongly recommended that you use a homogenous SRGB (meaning, the same SRGB range across all nodes). Using a heterogenous SRGB (meaning, a different SRGB range of the same size across nodes) is also supported but is not recommended.

### Behaviors and Limitations

- The default SRGB in IOS XR has a size of 8000 starting from label value 16000. The default range is 16000 to 23,999. With this size, and assuming one loopback prefix per router, an operator can assign prefix SIDs to a network with 8000 routers.
- There are instances when you might need to define a different SRGB range. For example:
  - Non-IOS XR nodes with a SRGB range that is different than the default IOS XR SRGB range.
  - The default SRGB range is not large enough to accommodate all required prefix SIDs.
- A non-default SRGB can be configured following these guidelines:
  - The SRGB starting value can be configured anywhere in the dynamic label range space (16,000 to 1,048,575).
  - In Cisco IOS XR release earlier than 6.6.3, the SRGB can have a maximum configurable size of 262,143.
  - In Cisco IOS XR release 6.6.3 and later, the SRGB can be configured to any size value that fits within the dynamic label range space.




---

**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.

---

- Allocating an SRGB label range does not mean that all the labels in this range are programmed in the forwarding table. The label range is just reserved for SR and not available for other purposes. Furthermore, a platform may limit the number of local labels that can be programmed.
- We recommend that the non-default SRGB be configured under the **segment-routing** global configuration mode. By default, all IGP instances and BGP use this SRGB.
- You can also configure a non-default SRGB under the IGP, but it is not recommended.

### SRGB Label Conflicts

When you define a non-default SRGB range, there might be a label conflict (for example, if labels are already allocated, statically or dynamically, in the new SRGB range). The following system log message indicates a label conflict:

```
%ROUTING-ISIS-4-SRGB_ALLOC_FAIL : SRGB allocation failed: 'SRGB reservation not
successful for [16000,80000], SRGB (16000 80000, SRGB_ALLOC_CONFIG_PENDING, 0x2)
(So far 16 attempts). Make sure label range is free'
```

To remove this conflict, you must reload the router to release the currently allocated labels and to allocate the new SRGB.

After the system reloads, LSD does not accept any dynamic label allocation before IS-IS/OSPF/BGP have registered with LSD. Upon IS-IS/OSPF/BGP registration, LSD allocates the requested SRGB (either the default range or the customized range).

After IS-IS/OSPF/BGP have registered and their SRGB is allocated, LSD starts serving dynamic label requests from other clients.



**Note** To avoid a potential router reload due to label conflicts, and assuming that the default SRGB size is large enough, we recommend that you use the default IOS XR SRGB range.



**Note** Allocating a non-default SRGB in the upper part of the MPLS label space increases the chance that the labels are available and a reload can be avoided.



**Caution** Modifying a SRGB configuration is disruptive for traffic and may require a reboot if the new SRGB is not available entirely.

## 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. **segment-routing global-block** *starting\_value ending\_value*
3. Use the **commit** or **end** command.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# <code>configure</code>	Enters mode.
Step 2	<b>segment-routing global-block</b> <i>starting_value ending_value</i> <b>Example:</b> RP/0/RP0/CPU0:router(config)# <code>segment-routing global-block 16000 80000</code>	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.

	Command or Action	Purpose
<b>Step 3</b>	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

Use the **show mpls label table [label label-value]** command to verify the SRGB configuration:

```
Router# show mpls label table label 16000 detail
Table Label  Owner                               State Rewrite
-----
0      16000  ISIS(A):1                                       InUse  No
      (Lbl-blk SRGB, vers:0, (start_label=16000, size=64001))
```

### What to do next

Configure prefix SIDs and enable segment routing.



## CHAPTER 5

# Configure Segment Routing for IS-IS Protocol

Integrated Intermediate System-to-Intermediate System (IS-IS), Internet Protocol Version 4 (IPv4), is a standards-based Interior Gateway Protocol (IGP). The Cisco IOS XR software implements the IP routing capabilities described in International Organization for Standardization (ISO)/International Engineering Consortium (IEC) 10589 and RFC 1995, and adds the standard extensions for single topology and multitopology IS-IS for IP Version 6 (IPv6).

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



**Note** 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*.

- [Enabling Segment Routing for IS-IS Protocol, on page 13](#)
- [Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface, on page 15](#)
- [IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability, on page 17](#)
- [IS-IS Multi-Domain Prefix SID and Domain Stitching: Example, on page 20](#)

## 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.



**Note** 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. Use the **commit** or **end** command.

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# <b>configure</b>	Enters mode.
<b>Step 2</b>	<b>router isis <i>instance-id</i></b> <b>Example:</b> RP/0/RP0/CPU0:router(config)# <b>router isis isp</b>	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.  <b>Note</b> You can change the level of routing to be performed by a particular routing instance by using the <b>is-type</b> router configuration command.
<b>Step 3</b>	<b>address-family ipv4 [ unicast ]</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-isis)# <b>address-family ipv4 unicast</b>	Specifies the IPv4 address family, and enters router address family configuration mode.
<b>Step 4</b>	<b>metric-style wide [ level { 1   2 }]</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-isis-af)# <b>metric-style wide level 1</b>	Configures a router to generate and accept only wide link metrics in the Level 1 area.
<b>Step 5</b>	<b>segment-routing mpls</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-isis-af)# <b>segment-routing mpls</b>	Segment routing is enabled by the following actions: <ul style="list-style-type: none"> <li>• MPLS forwarding is enabled on all interfaces where IS-IS is active.</li> <li>• All known prefix-SIDs in the forwarding plain are programmed, with the prefix-SIDs advertised by</li> </ul>

	Command or Action	Purpose
		<p>remote routers or learned through local or remote mapping server.</p> <ul style="list-style-type: none"> <li>• The prefix-SIDs locally configured are advertised.</li> </ul>
Step 6	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config-isis-af)# exit RP/0/RP0/CPU0:router(config-isis)# exit</pre>	
Step 7	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

### 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. Use the **commit** or **end** command.

**DETAILED STEPS**

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# <b>configure</b>	Enters mode.
<b>Step 2</b>	<b>router isis</b> <i>instance-id</i> <b>Example:</b> RP/0/RP0/CPU0:router(config)# <b>router isis</b> 1	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode. <ul style="list-style-type: none"> <li>• You can change the level of routing to be performed by a particular routing instance by using the <b>is-type</b> router configuration command.</li> </ul>
<b>Step 3</b>	<b>interface Loopback</b> <i>instance</i> <b>Example:</b> RP/0/RP0/CPU0:router(config-isis)# <b>interface Loopback0</b>	Specifies the loopback interface and instance.
<b>Step 4</b>	<b>address-family ipv4</b> [ <b>unicast</b> ] <b>Example:</b> The following is an example for ipv4 address family: RP/0/RP0/CPU0:router(config-isis-if)# <b>address-family ipv4 unicast</b>	Specifies the IPv4 address family, and enters router address family configuration mode.
<b>Step 5</b>	Use the <b>commit</b> or <b>end</b> command.	<b>commit</b> —Saves the configuration changes and remains within the configuration session. <b>end</b> —Prompts user to take one of these actions: <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> </ul>



	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

Verify the prefix-SID 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)
  Hostname:     router
  IP Address:   10.0.0.1
  Router Cap:   10.0.0.1, D:0, S:0
  Segment Routing: I:1 V:1, SRGB Base: 16000 Range: 8000
<...>
  Metric: 0          IP-Extended 10.0.0.1/32
  Prefix-SID Index: 1001, Algorithm:0, R:0 N:1 P:0 E:0 V:0 L:0
<...>
```

## 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.



**Note** Prefix attributes are only added when wide metric is used.

#### Prefix Attribute Flags Sub-TLV Format

```
0 1 2 3 4 5 6 7 ...
+---+---+---+---+---+---+---+---+...
```



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. Use the **commit** or **end** command.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure</b> <b>Example:</b>  RP/0/RP0/CPU0:router# <b>configure</b>	Enters mode.
Step 2	<b>router isis</b> <i>instance-id</i> <b>Example:</b>  RP/0/RP0/CPU0:router(config)# <b>router isis 1</b>	
Step 3	<b>interface Loopback</b> <i>instance</i> <b>Example:</b>  RP/0/RP0/CPU0:router(config)# <b>interface Loopback0</b>	Specifies the loopback interface.
Step 4	<b>prefix-attributes n-flag-clear</b> [Level-1   Level-2] <b>Example:</b>	Clears the prefix attribute N-flag explicitly.

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-if)# <b>isis prefix-attributes n-flag-clear</b>	
<b>Step 5</b>	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

Verify the prefix attribute configuration:

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

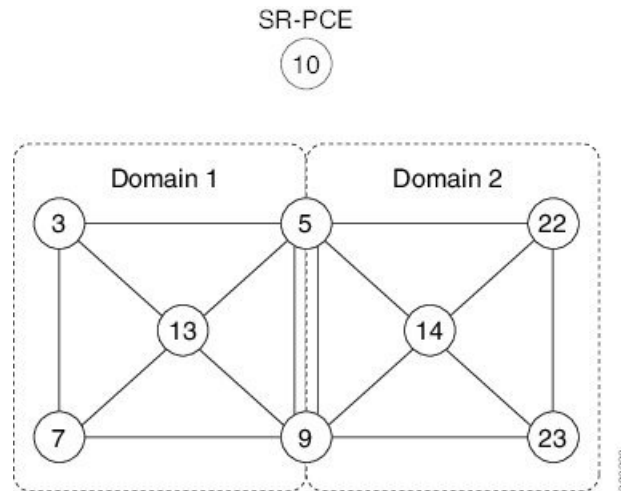
IS-IS 1 (Level-2) Link State Database
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
router.00-00        * 0x0000039b  0xfc27        1079          0/0/0
  Area Address: 49.0001
  NLPID:           0xcc
  NLPID:           0x8e
  MT:             Standard (IPv4 Unicast)
  MT:             IPv6 Unicast
  Hostname:       router
  IP Address:     10.0.0.1
  IPv6 Address:   2001:0db8:1234::0a00:0001
  Router Cap:    10.0.0.1, D:0, S:0
  Segment Routing: I:1 V:1, SRGB Base: 16000 Range: 8000
  SR Algorithm:
    Algorithm: 0
    Algorithm: 1
<...>
Metric: 0           IP-Extended 10.0.0.1/32
  Prefix-SID Index: 1001, Algorithm:0, R:1 N:0 P:1 E:0 V:0 L:0
  Prefix Attribute Flags: X:0 R:1 N:0
Metric: 10          IP-Extended 10.0.0.2/32
  Prefix-SID Index: 1002, Algorithm:0, R:0 N:1 P:0 E:0 V:0 L:0
  Prefix Attribute Flags: X:0 R:0 N:1
  Source Router ID: 10.0.0.2
<...>
```

## 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
router isis Domain1
 interface Loopback0
  address-family ipv4 unicast
  prefix-sid absolute 16005

router isis Domain2
 interface Loopback0
  address-family ipv4 unicast
  prefix-sid absolute 16005

```

```

Example: Border Node 9
router isis Domain1
 interface Loopback0
  address-family ipv4 unicast
  prefix-sid absolute 16009

router isis Domain2
 interface Loopback0
  address-family ipv4 unicast
  prefix-sid absolute 16009

```

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**

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

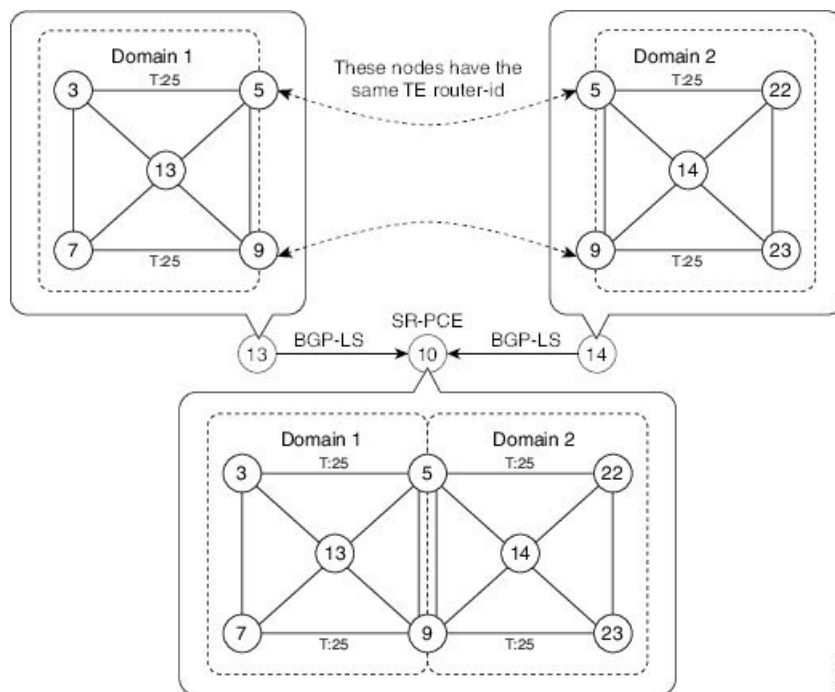
```
router isis Domain2
 address-family ipv4 unicast
  router-id loopback0
```

**Example: Border Node 9**

```
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



Configure BGP Link-state (BGP-LS) on Node 13 and Node 14 to report their local domain to Node 10:

**Example: Node 13**

```
router isis Domain1
 distribute link-state instance-id instance-id
```

**Example: Node 14**

```
router isis Domain2
  distribute link-state instance-id instance-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.







## CHAPTER 6

# 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 , see the *Implementing OSPF* module in the .

- [Enabling Segment Routing for OSPF Protocol, on page 25](#)
- [Configuring a Prefix-SID on the OSPF-Enabled Loopback Interface, on page 27](#)
- [Configuring an Adjacency SID, on page 29](#)

## 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.



**Note** 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** *area*
5. **segment-routing mpls**
6. **exit**
7. Use the **commit** or **end** command.

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b>  RP/0/RP0/CPU0:router# <b>configure</b>	Enters mode.
<b>Step 2</b>	<b>router ospf</b> <i>process-name</i> <b>Example:</b>  RP/0/RP0/CPU0:router(config)# <b>router ospf 1</b>	Enables OSPF routing for the specified routing process and places the router in router configuration mode.
<b>Step 3</b>	<b>segment-routing mpls</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-ospf)# <b>segment-routing mpls</b>	Enables segment routing using the MPLS data plane on the routing process and all areas and interfaces in the routing process.  Enables segment routing forwarding on all interfaces in the routing process and installs the SIDs received by OSPF in the forwarding table.
<b>Step 4</b>	<b>area</b> <i>area</i> <b>Example:</b>  RP/0/RP0/CPU0:router(config-ospf)# <b>area 0</b>	Enters area configuration mode.
<b>Step 5</b>	<b>segment-routing mpls</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-ospf-ar)# <b>segment-routing mpls</b>	(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.
<b>Step 6</b>	<b>exit</b> <b>Example:</b>	

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-ospf-ar)# <b>exit</b> RP/0/RP0/CPU0:router(config-ospf)# <b>exit</b>	
<b>Step 7</b>	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

**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. Use the **commit** or **end** command.

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# <b>configure</b>	Enters mode.
<b>Step 2</b>	<b>router ospf</b> <i>process-name</i> <b>Example:</b> RP/0/RP0/CPU0:router (config)# <b>router ospf 1</b>	Enables OSPF routing for the specified routing process, and places the router in router configuration mode.
<b>Step 3</b>	<b>area</b> <i>value</i> <b>Example:</b> RP/0/RP0/CPU0:router (config-ospf)# <b>area 0</b>	Enters area configuration mode.
<b>Step 4</b>	<b>interface Loopback</b> <i>interface-instance</i> <b>Example:</b> RP/0/RP0/CPU0:router (config-ospf-ar)# <b>interface Loopback0 passive</b>	Specifies the loopback interface and instance.
<b>Step 5</b>	Use the <b>commit</b> or <b>end</b> command.	<b>commit</b> —Saves the configuration changes and remains within the configuration session. <b>end</b> —Prompts user to take one of these actions: <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

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

```

## Configuring an Adjacency SID

An adjacency SID (Adj-SID) is associated with an adjacency to a neighboring node. The adjacency SID steers the traffic to a specific adjacency. Adjacency SIDs have local significance and are only valid on the node that allocates them.

An adjacency SID can be allocated dynamically from the dynamic label range or configured manually from the segment routing local block (SRLB) range of labels.

Adjacency SIDs that are dynamically allocated do not require any special configuration, however there are some limitations:

- A dynamically allocated Adj-SID value is not known until it has been allocated, and a controller will not know the Adj-SID value until the information is flooded by the IGP.
- Dynamically allocated Adj-SIDs are not persistent and can be reallocated after a reload or a process restart.
- Each link is allocated a unique Adj-SID, so the same Adj-SID cannot be shared by multiple links.

Manually allocated Adj-SIDs are persistent over reloads and restarts. They can be provisioned for multiple adjacencies to the same neighbor or to different neighbors. You can specify that the Adj-SID is protected. If the Adj-SID is protected on the primary interface and a backup path is available, a backup path is installed. By default, manual Adj-SIDs are not protected.

Adjacency SIDs are advertised using the existing OSPF Adj-SID sub-TLV. The P-flag is defined for manually allocated Adj-SIDs.

```

 0 1 2 3 4 5 6 7
+-+-----+-----+-----+
|B|V|L|G|P|   |
+-+-----+-----+-----+

```

**Table 2: Adjacency Segment Identifier (Adj-SID) Flags Sub-TLV Fields**

Field	Description
P (Persistent)	This flag is set if the Adj-SID is persistent (manually allocated).

This task explains how to configure an Adj-SID on an interface.

### Before you begin

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

Use the **show mpls label table detail** command to verify the SRLB range.

## SUMMARY STEPS

1. **configure**
2. **router ospf** *process-name*
3. **area** *area*
4. **interface** *type interface-path-id*
5. **adjacency-sid** {**index** *adj-SID-index* | **absolute** *adj-SID-value*} [**protected**]
6. Use the **commit** or **end** command.

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b>  RP/0/RP0/CPU0:router# <b>configure</b>	Enters mode.
<b>Step 2</b>	<b>router ospf</b> <i>process-name</i> <b>Example:</b>  RP/0/RP0/CPU0:router (config)# <b>router ospf</b> 1	Enables OSPF routing for the specified routing instance, and places the router in router configuration mode.
<b>Step 3</b>	<b>area</b> <i>area</i> <b>Example:</b>  RP/0/RP0/CPU0:router (config-ospf)# <b>area</b> 0	Enters area configuration mode.
<b>Step 4</b>	<b>interface</b> <i>type interface-path-id</i> <b>Example:</b>  RP/0/RP0/CPU0:router (config-ospf-ar)# <b>interface</b> <b>HundredGigE0/0/0/1</b>	Specifies the interface and enters interface configuration mode.
<b>Step 5</b>	<b>adjacency-sid</b> { <b>index</b> <i>adj-SID-index</i>   <b>absolute</b> <i>adj-SID-value</i> } [ <b>protected</b> ] <b>Example:</b>  RP/0/RP0/CPU0:router (config-config-ospf-ar-if)# <b>adjacency-sid index</b> 10  RP/0/RP0/CPU0:router (config-config-ospf-ar-if)# <b>adjacency-sid absolute</b> 15010	Configures the Adj-SID index or absolute value for the interface.  Specify <b>index</b> <i>adj-SID-index</i> for each link to create an Adj-SID based on the lower boundary of the SRLB + the index.  Specify <b>absolute</b> <i>adj-SID-value</i> for each link to create a specific Adj-SID within the SRLB.  Specify if the Adj-SID is <b>protected</b> . For each primary path, if the Adj-SID is protected on the primary interface and a backup path is available, a backup path is installed. By default, manual Adj-SIDs are not protected.
<b>Step 6</b>	Use the <b>commit</b> or <b>end</b> command.	<b>commit</b> —Saves the configuration changes and remains within the configuration session.  <b>end</b> —Prompts user to take one of these actions:

	Command or Action	Purpose
		<ul style="list-style-type: none"><li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li><li>• <b>No</b> — Exits the configuration session without committing the configuration changes.</li><li>• <b>Cancel</b> — Remains in the configuration session, without committing the configuration changes.</li></ul>







## CHAPTER 7

# 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.



---

**Note** For additional information on implementing BGP on your router, see the *BGP Configuration Guide for Cisco NCS 5000 Series Routers*.

---

- [Segment Routing for BGP, on page 33](#)
- [Configure BGP Prefix Segment Identifiers, on page 34](#)
- [Segment Routing Egress Peer Engineering, on page 35](#)
- [Configure BGP Link-State, on page 38](#)
- [Use Case: Configuring SR-EPE and BGP-LS, on page 40](#)

## 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 10.1.1.1
RP/0/RSP0/CPU0:router(config-bgp)# address-family ipv4 unicast
RP/0/RSP0/CPU0:router(config-bgp-af)# network 10.1.1.3/32 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-bgp-af)# end

RP/0/RSP0/CPU0:router# show bgp 10.1.1.3/32
BGP routing table entry for 10.1.1.3/32
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
```

```
Path #1: Received by speaker 0
Advertised to update-groups (with more than one peer):
  0.2
  3
    99.3.21.3 from 99.3.21.3 (10.1.1.3)
      Received Label 3
      Origin IGP, metric 0, localpref 100, valid, external, best, group-best
      Received Path ID 0, Local Path ID 1, version 74
      Origin-AS validity: not-found
      Label Index: 3
```

## 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.

Below are the BGP-EPE peering SID types:

- PeerNode SID—To an eBGP peer. Pops the label and forwards the traffic on any interface to the peer.
- PeerAdjacency SID—To an eBGP peer via interface. Pops the label and forwards the traffic on the related interface.

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.

## Configure Segment Routing Egress Peer Engineering

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**
5. **exit**
6. Use the **commit** or **end** command.

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<p><b>router</b> <i>bgp as-number</i></p> <p><b>Example:</b></p> <pre>RP/0/RSP0/CPU0:router(config)# <b>router bgp 1</b></pre>	Specifies the BGP AS number and enters the BGP configuration mode, allowing you to configure the BGP routing process.
<b>Step 2</b>	<p><b>neighbor</b> <i>ip-address</i></p> <p><b>Example:</b></p> <pre>RP/0/RSP0/CPU0:router(config-bgp)# <b>neighbor 192.168.1.3</b></pre>	Places the router in neighbor configuration mode for BGP routing and configures the neighbor IP address as a BGP peer.
<b>Step 3</b>	<p><b>remote-as</b> <i>as-number</i></p> <p><b>Example:</b></p> <pre>RP/0/RSP0/CPU0:router(config-bgp-nbr)# <b>remote-as 3</b></pre>	Creates a neighbor and assigns a remote autonomous system number to it.
<b>Step 4</b>	<p><b>egress-engineering</b></p> <p><b>Example:</b></p> <pre>RP/0/RSP0/CPU0:router(config-bgp-nbr)# <b>egress-engineering</b></pre>	Configures the egress node with EPE for the eBGP peer.
<b>Step 5</b>	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>RP/0/RSP0/CPU0:router(config-bgp-nbr)# <b>exit</b> RP/0/RSP0/CPU0:router(config-bgp)# <b>exit</b> RP/0/RSP0/CPU0:router(config)#</pre>	
<b>Step 6</b>	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

### Example

#### Running Config:

```
router bgp 1
 neighbor 192.168.1.3
   remote-as 3
   egress-engineering
 !
 !
 !
```

## 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.



**Note** You must reload the router after enabling ECMP, else the router may not function as expected.

```
RP/0/RP0/CPU0:router(config)# hw-module loadbalancing bgp-3107 ecmp enable  
RP/0/RP0/CPU0:router(config)# commit  
RP/0/RP0/CPU0:router(config)# end  
RP/0/RP0/CPU0:router(config)# reload
```

You have successfully enabled ECMP for BGP LU.

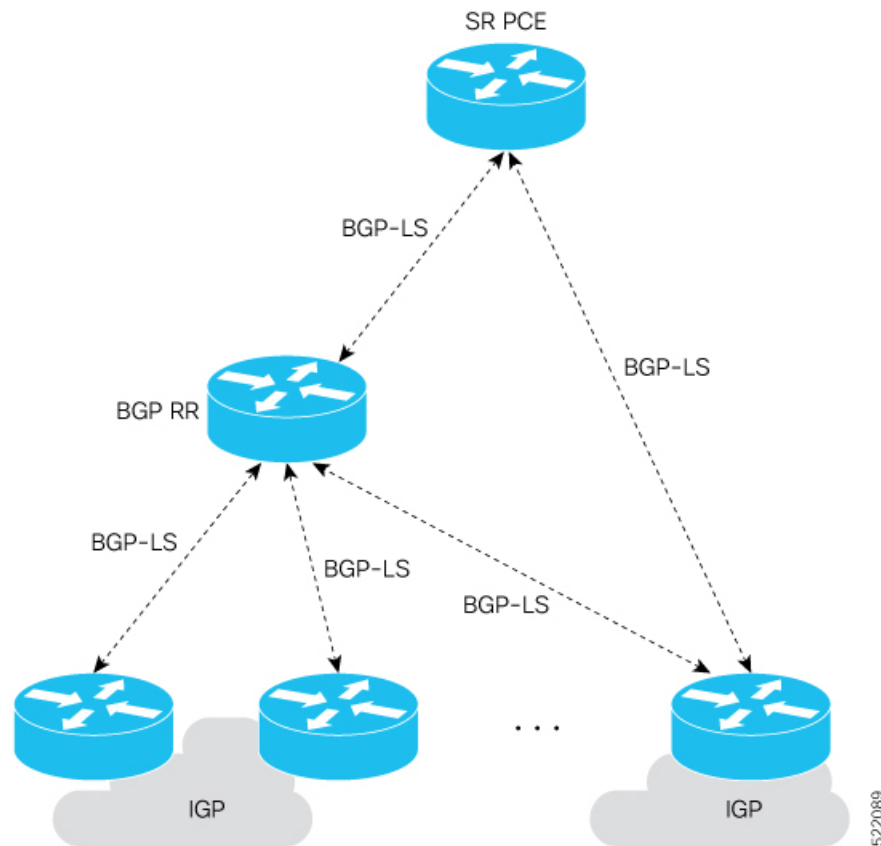
## Configure BGP Link-State

BGP Link-State (LS) is an Address Family Identifier (AFI) and Sub-address Family Identifier (SAFI) originally defined to carry interior gateway protocol (IGP) link-state information through BGP. The BGP Network Layer Reachability Information (NLRI) encoding format for BGP-LS and a new BGP Path Attribute called the BGP-LS attribute are defined in [RFC7752](#). The identifying key of each Link-State object, namely a node, link, or prefix, is encoded in the NLRI and the properties of the object are encoded in the BGP-LS attribute.

The BGP-LS Extensions for Segment Routing are documented in [RFC9085](#).

BGP-LS applications like an SR Path Computation Engine (SR-PCE) can learn the SR capabilities of the nodes in the topology and the mapping of SR segments to those nodes. This can enable the SR-PCE to perform path computations based on SR-TE and to steer traffic on paths different from the underlying IGP-based distributed best-path computation.

The following figure shows a typical deployment scenario. In each IGP area, one or more nodes (BGP speakers) are configured with BGP-LS. These BGP speakers form an iBGP mesh by connecting to one or more route-reflectors. This way, all BGP speakers (specifically the route-reflectors) obtain Link-State information from all IGP areas (and from other ASes from eBGP peers).



### Usage Guidelines and Limitations

- BGP-LS supports IS-IS and OSPFv2.
- The identifier field of BGP-LS (referred to as the Instance-ID) identifies the IGP routing domain where the NLRIs belong. The NLRIs representing link-state objects (nodes, links, or prefixes) from the same IGP routing instance must use the same Instance-ID value.
- When there is only a single protocol instance in the network where BGP-LS is operational, we recommend configuring the Instance-ID value to **0**.
- Assign consistent BGP-LS Instance-ID values on all BGP-LS Producers within a given IGP domain.
- NLRIs with different Instance-ID values are considered to be from different IGP routing instances.
- Unique Instance-ID values must be assigned to routing protocol instances operating in different IGP domains. This allows the BGP-LS Consumer (for example, SR-PCE) to build an accurate segregated multi-domain topology based on the Instance-ID values, even when the topology is advertised via BGP-LS by multiple BGP-LS Producers in the network.
- If the BGP-LS Instance-ID configuration guidelines are not followed, a BGP-LS Consumer may see duplicate link-state objects for the same node, link, or prefix when there are multiple BGP-LS Producers deployed. This may also result in the BGP-LS Consumers getting an inaccurate network-wide topology.

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

### Exchange Link State Information with BGP Neighbor

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

```
Router# configure
Router(config)# router bgp 1
Router(config-bgp)# neighbor 10.0.0.2
Router(config-bgp-nbr)# remote-as 1
Router(config-bgp-nbr)# address-family link-state link-state
Router(config-bgp-nbr-af)# exit
```

### IGP Link-State Database Distribution

A given BGP node may have connections to multiple, independent routing domains. IGP link-state database distribution into BGP-LS is supported for both OSPF and IS-IS protocols in order to distribute this information on to controllers or applications that desire to build paths spanning or including these multiple domains.

To distribute IS-IS link-state data using BGP-LS, use the **distribute link-state** command in router configuration mode.

```
Router# configure
Router(config)# router isis isp
Router(config-isis)# distribute link-state instance-id 32
```

To distribute OSPFv2 link-state data using BGP-LS, use the **distribute link-state** command in router configuration mode.

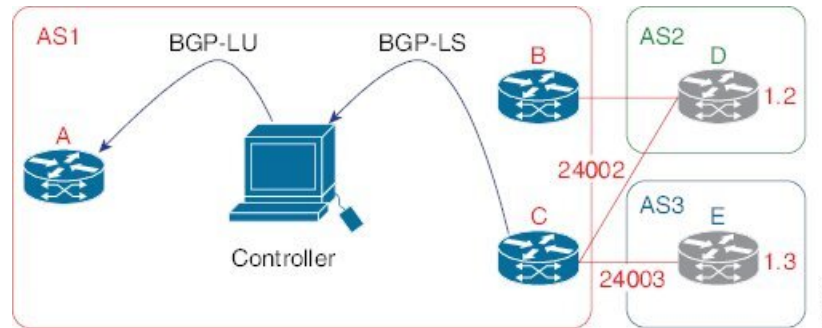
```
Router# configure
Router(config)# router ospf 100
Router(config-ospf)# distribute link-state instance-id 32
```

## Use Case: 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:**

```
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: 10.1.1.3
  Remote RID: 10.1.1.4
  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: 10.1.1.3
  Remote RID: 10.1.1.5
  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
Local  Outgoing  Prefix      Outgoing    Next Hop    Bytes
Label  Label      or ID      Interface   Interface   Switched
-----
24002  Pop        No ID      Te0/3/0/0   192.168.1.2  0
24003  Pop        No ID      Te0/1/0/0   192.168.1.3  0
```

The output shows that node C installed peer node SIDs in the Forwarding Information Base (FIB).



## CHAPTER 8

# 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.

- Classic Loop-Free Alternate (LFA) is topology dependent, and therefore cannot protect all destinations in all networks. A limitation of LFA is that, even if one or more LFAs exist, the optimal LFA may not always be provided.
- Remote LFA (RLFA) extends the coverage to 90-95% of the destinations, but it also does not always provide the most desired repair path. RLFA also adds more operational complexity by requiring a targeted LDP session to the RLFAs to protect LDP traffic.

TI-LFA provides a solution to these limitations while maintaining the simplicity of the IPFRR solution.

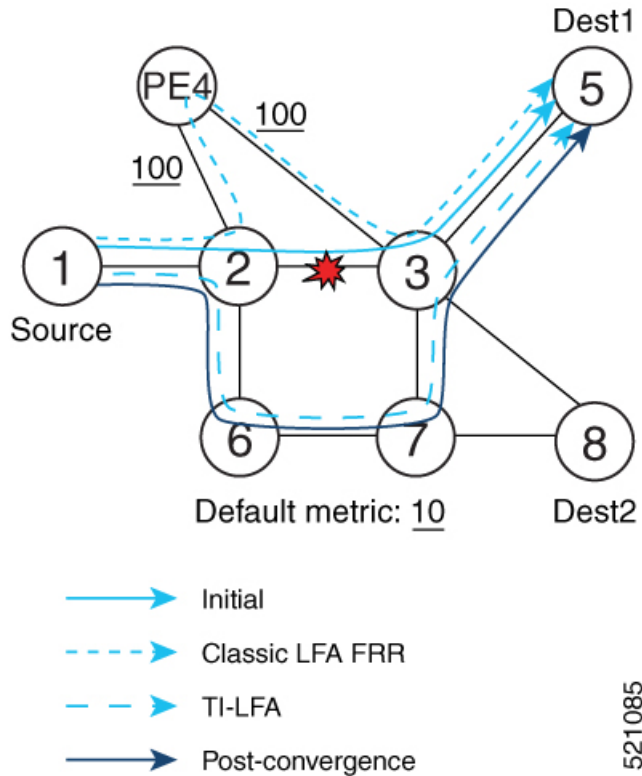
The goal of TI-LFA is to reduce the packet loss that results while routers converge after a topology change due to a link failure. Rapid failure repair (< 50 msec) is achieved through the use of pre-calculated backup paths that are loop-free and safe to use until the distributed network convergence process is completed.

The optimal repair path is the path that the traffic will eventually follow after the IGP has converged. This is called the post-convergence path. This path is preferred for the following reasons:

- Optimal for capacity planning — During the capacity-planning phase of the network, the capacity of a link is provisioned while taking into consideration that such link will be used when other links fail.
- Simple to operate — There is no need to perform a case-by-case adjustment to select the best LFA among multiple candidate LFAs.
- Fewer traffic transitions — Since the repair path is equal to the post-convergence path, the traffic switches paths only once.

The following topology illustrates the optimal and automatic selection of the TI-LFA repair path.

Figure 3: TI-LFA Repair Path



Node 2 protects traffic to destination Node 5.

With classic LFA, traffic would be steered to Node 4 after a failure of the protected link. This path is not optimal, since traffic is routed over edge node Node 4 that is connected to lower capacity links.

TI-LFA calculates a post-convergence path and derives the segment list required to steer packets along the post-convergence path without looping back.

In this example, if the protected link fails, the shortest path from Node2 to Node5 would be:

Node2 → Node6 → Node7 → Node3 → Node5

Node7 is the PQ-node for destination Node5. TI-LFA encodes a single segment (prefix SID of Node7) in the header of the packets on the repair path.

- [Usage Guidelines and Limitations, on page 44](#)
- [Configuring TI-LFA for IS-IS, on page 45](#)
- [Configuring TI-LFA for OSPF, on page 47](#)

## Usage Guidelines and Limitations

The TI-LFA guidelines and limitations are listed below:

TI-LFA Functionality	IS-IS <sup>1</sup>	OSPFv2
<i>Protected Traffic Types</i>		

TI-LFA Functionality	IS-IS <sup>1</sup>	OSPFv2
Protection for SR labeled traffic	Supported	Supported
Protection of IPv4 unlabeled traffic	Supported (IS-ISv4)	Supported
Protection of IPv6 unlabeled traffic	Unsupported	N/A
<b><i>Protection Types</i></b>		
Link Protection	Supported	Supported
Node Protection	Supported	Supported
Local SRLG Protection	Supported	Supported
Weighted Remote SRLG Protection		
Line Card Disjoint Protection	Supported	Unsupported
<b><i>Interface Types</i></b>		
Ethernet Interfaces	Supported	Supported
TI-LFA with L3VPN	Unsupported	Unsupported
Ethernet Bundle Interfaces	Supported	Supported
TI-LFA over GRE Tunnel as Protecting Interface		
<b><i>Additional Functionality</i></b>		
BFD-triggered		
BFDv6-triggered		N/A
Prefer backup path with lowest total metric	Supported	Supported
Prefer backup path from ECMP set	Supported	Supported
Prefer backup path from non-ECMP set	Supported	Supported
Load share prefixes across multiple backups paths	Supported	Supported
Limit backup computation up to the prefix priority	Supported	Supported

<sup>1</sup> Unless specified, IS-IS support is IS-ISv4 and IS-ISv6

## 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:

- Routers are configured with IS-IS.

- Segment routing for IS-IS is configured. See [Enabling Segment Routing for IS-IS Protocol, on page 13](#).

## 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

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

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-isis-if-af)# <b>fast-reroute per-prefix ti-lfa</b>	

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:

- Routers are configured with OSPF.
- Segment routing for OSPF is configured. See [Enabling Segment Routing for OSPF Protocol, on page 25](#).

### 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

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b>  RP/0/RP0/CPU0:router# <b>configure</b>	Enters mode.
<b>Step 2</b>	<b>router ospf</b> <i>process-name</i> <b>Example:</b>  RP/0/RP0/CPU0:router(config)# <b>router ospf 1</b>	Enables OSPF routing for the specified routing process, and places the router in router configuration mode.

	Command or Action	Purpose
<b>Step 3</b>	<b>area</b> <i>area-id</i> <b>Example:</b> RP/0/RP0/CPU0:router(config-ospf)# <b>area 1</b>	Enters area configuration mode.
<b>Step 4</b>	<b>interface</b> <i>type interface-path-id</i> <b>Example:</b> RP/0/RP0/CPU0:router(config-ospf-ar)# <b>interface</b> <b>GigabitEthernet0/0/2/1</b>	Enters interface configuration mode.
<b>Step 5</b>	<b>fast-reroute per-prefix</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-ospf-ar-if)# <b>fast-reroute per-prefix</b>	Enables per-prefix fast reroute.
<b>Step 6</b>	<b>fast-reroute per-prefix ti-lfa</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-ospf-ar-if)# <b>fast-reroute per-prefix ti-lfa</b>	Enables per-prefix TI-LFA fast reroute link protection.

TI-LFA has been successfully configured for segment routing.





## CHAPTER 9

# 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 49](#)
- [Segment Routing and LDP Interoperability, on page 50](#)
- [Configuring Mapping Server, on page 53](#)
- [Enable Mapping Advertisement, on page 55](#)
- [Enable Mapping Client, on page 57](#)

## 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.

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

## Segment Routing and LDP Interoperability

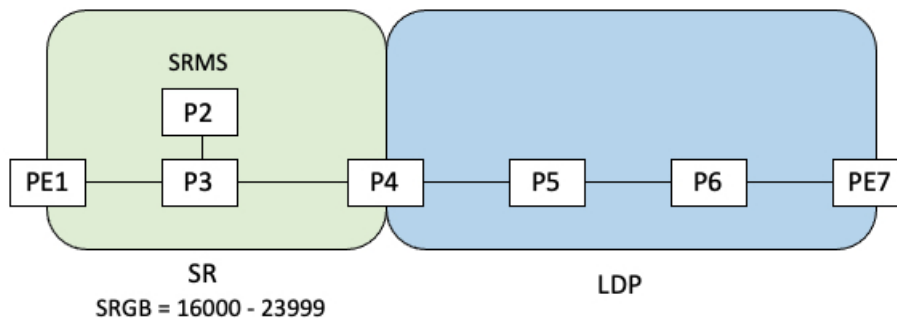
IGP provides mechanisms through which segment routing (SR) interoperates with label distribution protocol (LDP). The control plane of segment routing co-exists with LDP.

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 figure. This network is a mix of both LDP and SR-capable nodes.

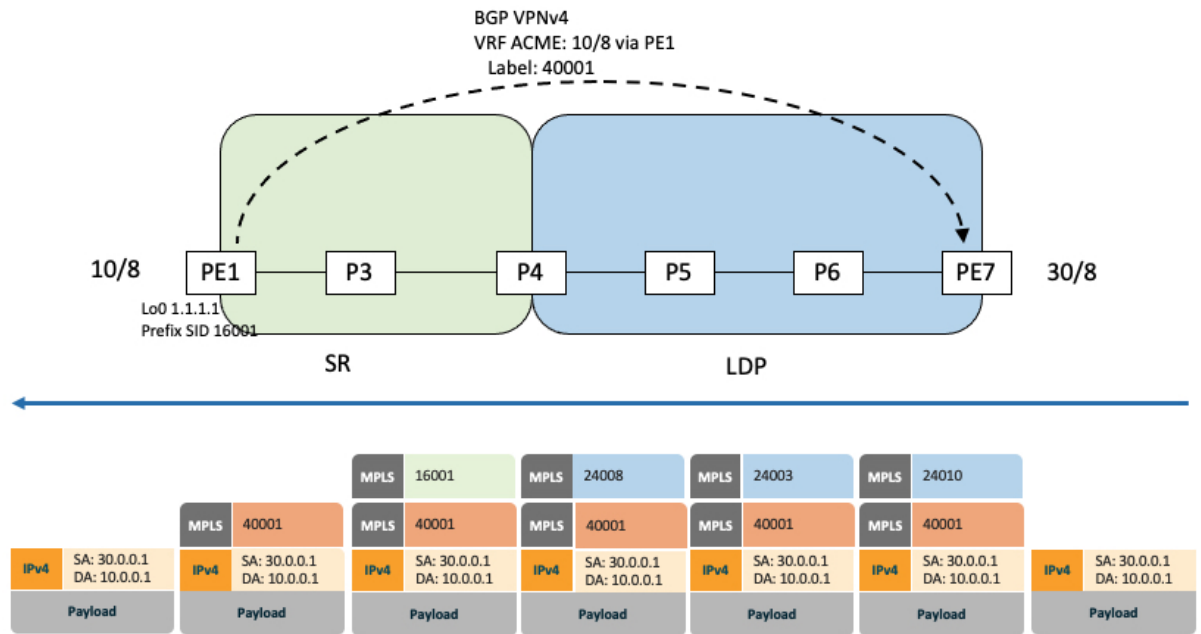


In this mixed network:

- Nodes PE1, P2, P3, and P4 are SR-capable
- Nodes P4, P5, P6, and PE7 are LDP-capable
- Nodes PE1, P2, P3, and P4 are configured with segment routing global block (SRGB) range of 16000 to 23999
- Nodes PE1, P2, P3, and P4 are configured with node segments of 16001, 16002, 16003, and 16004 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 Traffic Direction**



The traffic flow in the LDP-to-SR direction involves the following:

1. PE7 learns a service route with service label 40001 and BGP nhop PE1.
2. PE7 has an LDP label binding (24010) from the nhop P6 for the FEC PE1. PE7 forwards the packet to P6.
3. P6 has an LDP label binding (24003) from its nhop P5 for the FEC PE1. P6 forwards the packet to P5.
4. P5 has an LDP label binding (24008) from its nhop P4 for the FEC PE1. P5 forwards the packet to P4.
5. P4 does not have an LDP binding from its nhop P3 for the FEC PE1. But P4 has an SR node segment to the IGP route PE1. P4 forwards the packet to P3 and swaps its local LDP label (24008) for FEC PE1 by the equivalent node segment 16001. This process is called label merging.
6. P3 pops 16001, assuming PE1 has advertised its node segment 16001 with the penultimate-pop flag set and forwards to PE1.



The end-to-end MPLS LSP is established from an SR node segment from PE1 to P4 and an LDP LSP from P4 to PE7.

Observe that the capabilities provided by the SRMS are only required in the SR-to-LDP direction.

## 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. Use the **commit** or **end** command.

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b>  RP/0/RP0/CPU0:router# <code>configure</code>	Enters mode.
<b>Step 2</b>	<b>segment-routing</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config)# <code>segment-routing</code>	Enables segment routing.
<b>Step 3</b>	<b>mapping-server</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-sr)# <code>mapping-server</code>	Enables mapping server configuration mode.
<b>Step 4</b>	<b>prefix-sid-map</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-sr-ms)# <code>prefix-sid-map</code>	Enables prefix-SID mapping configuration mode.  <b>Note</b> Two-way prefix SID can be enabled directly under IS-IS or through a mapping server.
<b>Step 5</b>	<b>address-family ipv4  ipv6</b> <b>Example:</b> This example shows the address-family for ipv4:	Configures address-family for IS-IS.

	Command or Action	Purpose
	<pre>RP/0/RP0/CPU0:router(config-sr-ms-map)# address-family ipv4</pre> <p>This example shows the address-family for ipv6:</p> <pre>RP/0/RP0/CPU0:router(config-sr-ms-map)# address-family ipv6</pre>	
<b>Step 6</b>	<p><i>ip-address/prefix-length first-SID-value range range</i></p> <p><b>Example:</b></p> <pre>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</pre>	<p>Adds SID-mapping entries in the active local mapping policy. In the configured example:</p> <ul style="list-style-type: none"> <li>• 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</li> <li>• 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.</li> </ul>
<b>Step 7</b>	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

Verify information about the locally configured prefix-to-SID mappings.



**Note** Specify the address family for IS-IS.

```
RP/0/RP0/CPU0:router# show segment-routing mapping-server prefix-sid-map ipv4
Prefix          SID Index  Range  Flags
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. Use the `commit` or `end` command.

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>router isis <i>instance-id</i></code></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config)# router isis 1</pre>	<p>Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.</p> <ul style="list-style-type: none"> <li>You can change the level of routing to be performed by a particular routing instance by using the <b>is-type</b> router configuration command.</li> </ul>
Step 2	<p><code>address-family { ipv4   ipv6 } [ unicast ]</code></p> <p><b>Example:</b></p> <p>The following is an example for ipv4 address family:</p> <pre>RP/0/RP0/CPU0:router(config-isis)# address-family   ipv4 unicast</pre>	<p>Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.</p>
Step 3	<p><code>segment-routing prefix-sid-map advertise-local</code></p> <p><b>Example:</b></p>	<p>Configures IS-IS to advertise locally configured prefix-SID mappings.</p>

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-isis-af)# <b>segment-routing prefix-sid-map advertise-local</b>	
<b>Step 4</b>	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

Verify IS-IS prefix-SID mapping advertisement and TLV.

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

```
<...removed...>
```

```
SID Binding: 10.1.1.1/32 F:0 M:0 S:0 D:0 A:0 Weight:0 Range:200
SID: Start:10, Algorithm:0, R:0 N:0 P:0 E:0 V:0 L:0
SID Binding: 20.1.1.0/24 F:0 M:0 S:0 D:0 A:0 Weight:0 Range:300
SID: Start:400, Algorithm:0, R:0 N:0 P:0 E:0 V:0 L:0
```

## Configure Mapping Advertisement for OSPF

### SUMMARY STEPS

1. **router ospf** *process-name*
2. **segment-routing prefix-sid-map advertise-local**
3. Use the **commit** or **end** command.

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<p><b>router ospf</b> <i>process-name</i></p> <p><b>Example:</b></p> <pre>RP/0/RP0/CPU0:router(config)# router ospf 1</pre>	Enables OSPF routing for the specified routing instance, and places the router in router configuration mode.
<b>Step 2</b>	<p><b>segment-routing prefix-sid-map advertise-local</b></p> <p><b>Example:</b></p>	Configures OSPF to advertise locally configured prefix-SID mappings.



	Command or Action	Purpose
	RP/0/RP0/CPU0:router (config-ospf) # <b>segment-routing prefix-sid-map advertise-local</b>	
<b>Step 3</b>	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

Verify OSP prefix-SID mapping advertisement and TLV.

```
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
```

```
Flags       : 0x60
MTID        : 0
Algo        : 0
SID Index   : 10
```

## 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
```



## CHAPTER 10

# 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 59](#)
- [Examples: MPLS Ping, Traceroute, and Tree Trace for Prefix-SID, on page 60](#)
- [MPLS LSP Ping and Traceroute Nil FEC Target, on page 61](#)
- [Examples: LSP Ping and Traceroute for Nil\\_FEC Target, on page 62](#)

## 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.

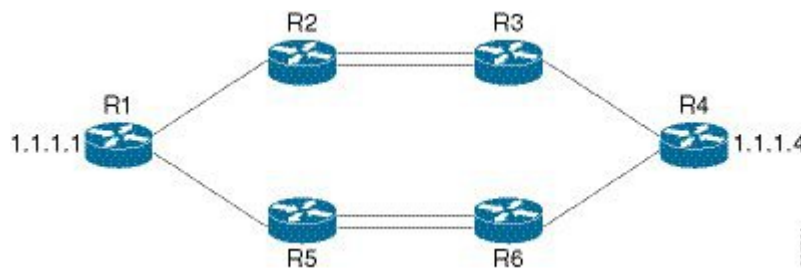
The MPLS LSP Tree Trace (traceroute multipath) operation is also supported for IGP Prefix SID. MPLS LSP Tree Trace provides the means to discover all possible equal-cost multipath (ECMP) routing paths of an LSP to reach a destination Prefix SID. It uses multipath data encoded in echo request packets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

load-balancing information that may allow the originator to exercise each ECMP. When the packet TTL expires at the responding node, the node returns the list of downstream paths, as well as the multipath information that can lead the operator to exercise each path in the MPLS echo reply. This operation is performed repeatedly for each hop of each path with increasing TTL values until all ECMP are discovered and validated.

MPLS echo request packets carry Target FEC Stack sub-TLVs. The Target FEC sub-TLVs are used by the responder for FEC validation. The IGP IPv4 prefix sub-TLV has been added to the Target FEC Stack sub-TLV. The IGP IPv4 prefix sub-TLV contains the prefix SID, the prefix length, and the protocol (IS-IS or OSPF).

## Examples: MPLS Ping, Traceroute, and Tree Trace for Prefix-SID

These examples use the following topology:



### MPLS Ping for Prefix-SID

```
RP/0/RP0/CPU0:router-arizona# ping mpls ipv4 10.1.1.4/32
Thu Dec 17 01:01:42.301 PST
```

```
Sending 5, 100-byte MPLS Echos to 10.1.1.4,
    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 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 10.1.1.4/32
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, '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.

 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 10.1.1.4/32
Thu Dec 17 14:55:46.549 PST
```

```
Starting LSP Path Discovery for 10.1.1.4/32
```

```
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 3: LSP Ping and Traceroute Nil FEC Commands**

Command Syntax
<b>ping mpls nil-fec labels</b> {label[,label]} [output {interface tx-interface} [nexthop nexthop-ip-addr]]
<b>traceroute mpls nil-fec labels</b> {label[,label]} [output {interface tx-interface} [nexthop nexthop-ip-addr]]

## 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:router-utah# show mpls forwarding
```

```
Tue Jul  5 13:44:31.999 EDT
Local  Outgoing  Prefix      Outgoing    Next Hop    Bytes
Label  Label      or ID       Interface   Interface   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:router-arizona# ping mpls nil-fec labels 16005,16007 output interface
GigabitEthernet 0/2/0/1 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 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,
       '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
```

### 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
```

```
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,
       'd' - see DDMAP for return code,
       'X' - unknown return code, 'x' - return code 0
```

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
```

