# Segment Routing Configuration Guide, Cisco IOS XE 17 | Cisco Catalyst 8000 Edge Platforms


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

# CONTENTS


CHAPTER 1 **Read Me First** 1

CHAPTER 2 **Introduction to Segment Routing** 3

- Feature Information for Introduction to Segment Routing 3
- Overview of Segment Routing 3
- How Segment Routing Works 4
- Examples for Segment Routing 5
- Benefits of Segment Routing 6
- Segment Routing Global Block 8
  - Segment Routing Global Block 9
  - Adjacency Segment Identifiers 9
  - Prefix Segment Identifiers 9
- Additional References for Segment Routing 10

CHAPTER 3 **Segment Routing With IS-IS v4 Node SID** 11

- Feature Information for Segment Routing—IS-IS v4 Node SID 11
- Restrictions for Segment Routing With IS-IS v4 Node SID 11
- Information About Segment Routing IS-IS v4 Node SID 12
  - Segment Routing IS-IS v4 Node SID 12
  - Prefix-SID Received in Label Switched Path from Remote Routers 12
  - Segment Routing Adjacency SID Advertisement 13
    - Multiple Adjacency-SIDs 13
  - Segment Routing Mapping Server (SRMS) 13
    - Connected Prefix SIDs 14
  - SRGB Range Changes 14
    - SRGB Deletion 14

MPLS Forwarding on an Interface 14
Segment Routing and LDP Preference 14
Segment Routing -Traffic Engineering Announcements 15
How to Configure Segment Routing —IS-IS v4 Node SID 15
Configuring Segment Routing 15
Configuring Segment Routing on IS-IS Network 16
Configuring Prefix-SID for IS-IS 17
Configuring Prefix Attribute N-flag-clear 19
Configuring Explicit Null Attribute 19
Configuring Segment Routing Label Distribution Protocol Preference 21
Configuring IS-IS SRMS 22
Configuring IS-IS SRMS Client 22
Configuring IS-IS SID Binding TLV Domain Flooding 22
Configuration Examples for Segment Routing —IS-IS v4 Node SID 22
Example: Configuring Segment Routing on IS-IS Network 22
Example: Configuring Explicit Null Attribute 23
Additional References for Segment Routing With IS-IS v4 Node SID 23

CHAPTER 4 **IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute 25**
Feature Information for IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute 25
Prerequisites for IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute 26
Information About IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute 27
Topology-Independent Loop Free Alternate 27
Topology Independent Loop Free Alternate Tie-break 28
Interface Fast Reroute Tiebreakers 28
How to Configure IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute 29
Configuring Topology Independent Loop Free Alternate Fast Reroute 29
Configuring Topology Independent Loop Free Alternate With Mapping Server 30
Examples: Configuring IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute 33
Verifying the Tiebreaker 35
Verifying the Primary and Repair Paths 35

Verifying the IS-IS Segment Routing Configuration 36
Verifying the IS-IS Topology Independent Loop Free Alternate Tunnels 37
Verifying the Segment Routing Traffic Engineering With Topology Independent Loop Free Alternate Configuration 37

CHAPTER 5 **Segment Routing Traffic Engineering With IS-IS 41**
Feature Information for Segment Routing -Traffic Engineering With IS-IS 41
Restrictions for Segment Routing-Traffic Engineering With IS-IS 41
Information About Segment Routing Traffic Engineering With IS-IS 42
SR-TE LSP Instantiation 42
SR-TE LSP Explicit Null 42
SR-TE LSP Path Verification 42
SR-TE Traffic Load Balancing 45
SR-TE Tunnel Re-optimization 45
SR-TE With Lockdown Option 46
SR-TE Tunnel Protection 47
Unnumbered Support 47
How to Configure Segment Routing Traffic Engineering With IS-IS 48
Configuring Path Option for a TE Tunnel 48
Configuring SR Explicit Path Hops 48
Configuring Affinity on an Interface 49
Enabling Verbatim Path Support 49
Use Case: Segment Routing Traffic Engineering Basic Configuration 49
Explicit Path SR-TE Tunnel 1 51
Explicit Path SR-TE Tunnel 2 52
Explicit Path SR-TE Tunnel 3 52
Dynamic Path SR-TE Tunnel 4 52
Dynamic Path SR-TE Tunnel 5 53
Verifying Configuration of the SR-TE Tunnels 53
Verifying Tunnel 1 53
Verifying Tunnel 2 54
Verifying Tunnel 3 54
Verifying Tunnel 4 55
Verifying Tunnel 5 55

Verifying Verbatim Path Support 56

CHAPTER 6 **Segment Routing With OSPFv2 Node SID 59**

Feature Information for Segment Routing With OSPFv2 Node SID 59

Information About Segment Routing With OSPFv2 Node SID 59

Prefix-SID Received in Label Switched Path From Remote Routers 60

Segment Routing Adjacency SID Advertisement 60

Multiple Adjacency-SIDs 61

Segment Routing Mapping Server 61

Connected Prefix SIDs 61

SRGB Range Changes 61

MPLS Forwarding on an Interface 61

Conflict Handling of SID Entries 62

How to Configure Segment Routing With OSPFv2 Node SID 62

Configuring Segment Routing With OSPF 62

Configuring Segment Routing on OSPF Network 63

Configuring Prefix-SID for OSPF 65

Configuring Prefix Attribute N-flag-clear 66

Configuring Explicit Null Attribute With OSPF 67

Configuring Segment Routing Label Distribution Protocol Preference With OSPF 68

Configuring OSPF SRMS 69

Configuring OSPF SRMS Client 69

Additional References for Segment Routing With OSPFv2 Node SID 70

CHAPTER 7 **OSPFv2 Link-protection Topology Independent Loop Free Alternate Fast Reroute 71**

Feature Information for OSPFv2 Link-protection Topology Independent Loop Free Alternate Fast Reroute 71

Restrictions for Topology Independent Loop Free Alternate Fast Reroute 72

Information About OSPFv2 Link-Protection Topology Independent Loop Free Alternate Fast Reroute 72

IP Fast Reroute and Remote Loop Free Alternate 73

Topology Independent Fast Reroute 73

Topology-Independent Loop Free Alternate 73

Topology Independent Loop Free Alternate Tie-break 74

P-Space 75
Q-Space 75
Post-Convergence Path 75
Per-Destination Link Protection 76
Per Interface Loop Free Alternate Enablement 76
Prefix Processing 76
Anycast Prefix Processing 77
Per-Prefix Loop Free Alternate Tie-Break 77
Node Protection 78
Shared Risk Link Groups Protection 79
Node-Shared Risk Link Groups Protection 79
How to Configure Topology Independent Loop Free Alternate Fast Reroute 80
Enabling Topology Independent Loop Free Alternate Fast Reroute 80
Configuring Topology Independent Loop Free Alternate Fast Reroute 80
Configuring Topology Independent Fast Reroute Tie-breaker 81
Verifying Topology Independent Fast Reroute Tunnels 83
Debugging Topology Independent Loop Free Alternate Fast Reroute 85
Examples: OSPFv2 Link-Protection Topology Independent Loop Free Alternate Fast Reroute 85
Example: Configuring Topology Independent Loop Free Alternate Fast Reroute 85

CHAPTER 8 **Segment Routing Traffic Engineering With OSPF 87**
Feature Information for Segment Routing Traffic Engineering With OSPF 87
Restrictions for Segment Routing Traffic Engineering With OSPF 88
Information About Segment Routing Traffic Engineering With OSPF 88
Benefits of Using Segment Routing Traffic Engineering With OSPF 88
OSPFv2 Segment Routing Traffic Engineering Functionalities 89
Protected Adjacency SID 89
Traffic Engineering Interfaces 89
Unnumbered Support 89
Segment Routing Traffic Engineering Support for Forwarding Adjacency 89
Segment Routing Traffic Engineering Support for Auto-route Announce 90
Auto-route Announce IP2MPLS 90
SR-TE LSP Instantiation 90
Tunnel Path Affinity Validation 90

SR-TE Traffic Load Balancing 91
Load Balancing on Port Channel TE Links 91
Load Balancing on Single Tunnel 91
Load Balancing on Multiple Tunnels 91
SR-TE Tunnel Re-optimization 91
SR-TE With Lockdown Option 92
SR-TE Tunnel Protection 93
IP-FRR Local Repair Protection 93
Tunnel Path Protection 93
SR-TE LSP Path Verification 94
Topology Path Validation 94
SR SID Validation 94
LSP Egress Interface 95
IP Reachability Validation 95
Tunnel Path Resource Avoidance Validation 95
SR-TE LSP Explicit Null 95
Verbatim Path Support 96
How to Configure Segment Routing Traffic Engineering With OSPF 96
Enabling Segment Routing Traffic Engineering With OSPF 96
Configuring Path Option for a TE Tunnel 96
Configuring SR Explicit Path Hops 97
Configuring Tunnel Path Affinity Validation 97
Configuring Affinity on an Interface 98
Configuring Segment Routing Traffic Engineering With OSPF 98
Configuring Intra Area Tunnel 98
Configuring Inter Area Tunnel 101
Verifying Configuration of the SR-TE Tunnels 104
Verifying Tunnel 1 104
Verifying Tunnel 2 104
Verifying Tunnel 3 105
Verifying Tunnel 4 106
Verifying Tunnel 5 106

CHAPTER 9 **BGP Dynamic Segment Routing Traffic Engineering** 109

Feature Information for BGP Dynamic Segment Routing Traffic Engineering 109
Restrictions for Segment Routing –Traffic-Engineering Dynamic BGP 109
Information About Segment Routing –Traffic-Engineering Dynamic BGP 110
TE Label Switched Path Attribute-Set 111
How to Configure TE Label Switched Path Attribute-Set 111
Configuring TE Label Switched Path Attribute-Set 111

CHAPTER 10 **Segment Routing On Demand Next Hop for L3/L3VPN 113**
Feature Information for Segment Routing On Demand Next Hop for L3/L3VPN 113
Restrictions for Segment Routing On Demand SR PFP ODN AUTO STEERING (PCE DELEGATED) for L3/L3VPN 114
Information About Segment Routing On Demand SR PFP ODN AUTO STEERING (PCE DELEGATED) for L3/L3VPN 114
SR-TE Policy, Color Extended Community, Affinity Constraint, and Disjointness Constraint 115
SR-TE Policy Command 115
Color Extended Community 115
Affinity Constraint 116
Disjointness Constraint 116
How to Configure Segment Routing On Demand Next Hop for L3/L3VPN 117
Configuring Segment Routing On Demand Next Hop for L3/L3VPN 117
Verifying Segment Routing On Demand Next Hop for L3/L3VPN 120
Configuring Color Extended Community, Affinity Constraint, and Disjointness Constraint 124
Configuring Color Extended Community 124
Configuring Affinity Constraint 125
Configuring Disjointness Constraint 126
Verifying SR-TE ODN Color Extended Community, Affinity Constraint, and Disjointness Constraint 126
Troubleshooting the SR-TE ODN Color Extended Community, Affinity Constraint, and Disjointness Constraint 131

CHAPTER 11 **Segment Routing On Demand for L2VPN/VPWS 135**
Feature Information for Segment Routing On Demand Next Hop for L2VPN/VPWS 135
Restrictions for Segment Routing On Demand Next Hop for L2VPN/VPWS 136
Information About Segment Routing On Demand Next Hop for L2VPN/VPWS 136

AToM Manager 137
Inter-Area L2VPN ODN 137
How to Configure Segment Routing On Demand Next Hop for L2VPN/VPWS 137
Configuring Segment Routing On Demand Next Hop for L2VPN/VPWS Using Pesudowire Interface Commands 137
Configuring Segment Routing On Demand Next Hop for L2VPN/VPWS Using Template Commands 138
Configuring Segment Routing On Demand Next Hop for L2VPN/VPWS With Prepend Option 139
Configuring Preferred Path for Segment Routing On Demand Next Hop for L2VPN/VPWS 139
Configuring Autoroute Destination for Segment Routing On Demand Next Hop for L2VPN/VPWS 140
Verifying Segment Routing On Demand Next Hop for L2VPN/VPWS 140

CHAPTER 12 **Fast Convergence Default Optimize** 145
Feature Information for Fast Convergence Default Optimize 145
Information About Fast Convergence Default Optimize 145
Default Optimize Values for IS-IS 146
Default Optimize Values for OSPF 147

CHAPTER 13 **Routing Information Base Support** 149
Feature Information for Routing Information Base Support 149
Routing Information Base Support for Route Redistribution 150
OSPF Node SID Redistribution Support 150
Information About OSPF Node SID Redistribution Support 150
NSSA ASBR 150
non-NSSA ASBR 150
Redistributing Prefix 151
Verify OSPF Node SID Redistribution 151
Routing Information Base Support for On-Demand Next Hop 152

CHAPTER 14 **SR-TE On Demand LSP** 155
Feature Information for SR-TE On Demand LSP 155
Restrictions for SR-TE On Demand LSP 155
Information About SR-TE On Demand LSP 156

SR-TE: Setup LSP as Static Route 156
Static SRTE over Unnumbered Interfaces 157
How to Configure SR-TE On Demand LSP 157
Configuring LSP as Static Route 157
Enabling Segment Routing Auto Tunnel Static Route 157
Verifying Segment Routing Auto-Tunnel Static Route 157

CHAPTER 15 **Segment Routing MPLS OAM Support 161**
Feature Information for Segment Routing OAM Support 161
Restrictions for Segment Routing OAM MPLS Support 162
Information About Segment Routing MPLS OAM Support 162
Segment Routing OAM Support 162
Benefits of Segment Routing OAM Support 163
Segment Routing MPLS Ping 163
Segment Routing MPLS Traceroute 163
LSP Ping Operation for Nil FEC target 164
How to Diagnose Segment Routing with LSP Ping and Trace Route Nil FEC Target 164
Using LSP Ping for Nil FEC Target 164
Using LSP Traceroute for Nil FEC Target 164
Example for LSP Ping Nil FEC Target Support 165
Path Validation in Segment Routing Network 166
MPLS Ping and Traceroute for IGP Prefix-SID FEC Type 167
MPLS Ping and Traceroute for IGP-Adjacency Segment ID 168
Configuring Segment Routing MPLS Traffic Engineering for MPLS Ping and Traceroute 168
Configuring Segment Routing MPLS IGP for MPLS Ping and Traceroute 169
Verifying Segment Routing OAM Using Cisco IOS CLI 170
Verifying Segment Routing Traffic Engineering OAM Operations 170
Verifying Segment Routing OAM OSPF Using CLI 171
Verifying Segment Routing OAM IS-IS Using CLI 174
Verifying MPLS Ping and Traceroute for IGP Segment ID 174

CHAPTER 16 **Using Seamless BFD with Segment Routing 175**
Feature Information for Seamless BFD with Segment Routing 175
Restrictions For Using Seamless BFD with Segment Routing 176

Information About Seamless BFD with Segment Routing 176
Bidirectional Forwarding Detection and Seamless-Bidirectional Forwarding Detection (S-BFD) 176
Initiators and Reflectors 176
How to Configure Seamless BFD with Segment Routing 177
Configuring Seamless-Bidirectional Forwarding Detection (S-BFD) for Segment Routing 177
Enabling Seamless Bidirectional Forwarding Detection (S-BFD) on the Reflector Node 178
Enabling Seamless Bidirectional Forwarding Detection (S-BFD) on the Initiator Node 178
Enabling Segment Routing Traffic Engineering Tunnel with Seamless-Bidirectional Forwarding (S-BFD) 178
Verifying S-BFD Configuration 178
Additional References for Seamless BFD with Segment Routing 179

CHAPTER 17 **Using SSPF with Segment Routing 181**
Feature Information for SSPF with Segment Routing 181
Information About SSPF with Segment Routing 181
Strict Shortest Path First 181
Approaches for Configure Strict Shortest Path First 182
How to Configure SSPF with Segment Routing 182
Configuring Strict Shortest Path First (SPF) 182
Enabling Strict Shortest Path First Using the connect-prefix-sid-map command 182
Enabling Strict Shortest Path First Using Segment Routing Mapping Server 183
Additional References for SSPF with Segment Routing 184

CHAPTER 18 **Dynamic PCC 185**
Information About Dynamic PCC 185
Path Computation Element Protocol Functions 185
Redundant Path Computation Elements 185
How to Configure Dynamic PCC 186
Configuring Dynamic PCC Globally 186
Configuring Dynamic PCC on an Interface 186
Configuring Dynamic PCC With Verbatim Path Option 186
Verifying Dynamic PCC 187
Verifying Verbatim Path Option With Dynamic PCC 190

Feature Information for Dynamic PCC 191

CHAPTER 19 SR: PCE Initiated LSPs 193

Prerequisites for SR: PCE Initiated LSPs 193

Restrictions for SR: PCE Initiated LSPs 193

Information About SR: PCE Initiated LSPs 193

Overview of Path Computation Element Protocol 193

SR: PCE Initiated LSPs 194

Single and Redundant PCE Operations 194

How to Configure SR: PCE Initiated LSPs 195

Establishing a PCEP session with PCC 195

Advertising an LSP in a Network 195

Specifying Precedence of a PCE for PCC 195

Verifying LSP Configurations 196

Additional References for SR: PCE Initiated LSPs 201

Feature Information for SR: PCE Initiated LSPs 201

CHAPTER 20 ISIS - SR: uLoop Avoidance 203

Prerequisites for ISIS - SR: uLoop Avoidance 203

Restrictions for ISIS - SR: uLoop Avoidance 203

Information About ISIS - SR: uLoop Avoidance 203

Microloops 203

Segment Routing and Microloops 206

How Segment Routing Prevents Microloops? 206

How to Enable ISIS - SR: uLoop Avoidance 207

Enabling Microloop Avoidance 207

Verifying Microloop Avoidance 207

Additional References for ISIS - SR: uLoop Avoidance 208

Feature Information for ISIS - SR: uLoop Avoidance 208

CHAPTER 21 BGP - SR: BGP Prefix SID Redistribution 211

Prerequisites for BGP - SR: BGP Prefix SID Redistribution 211

Information About BGP - SR: BGP Prefix SID Redistribution 211

Segment Routing and BGP 211

Segment Routing for Locally Sourced Routes 212
Segment Routing for Received Prefixes 212
Segment Routing for Redistributed Routes 212
BGP--MFI Interaction 212
How to Enable BGP - SR: BGP Prefix SID Redistribution 212
Enabling BGP-Prefix-SID 212
Enabling BGP for Segment Routing 213
Verifying BGP - SR: BGP Prefix SID Redistribution 213
Additional References for BGP - SR: BGP Prefix SID Redistribution 214
Feature Information for BGP - SR: BGP Prefix SID Redistribution 214

CHAPTER 22 **Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS** 215
Restrictions for Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS 215
Information About Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS 215
Maximum SID Depth 215
Node Maximum SID Depth Advertisement 216
Getting the Node MSD from Hardware 217
Advertising the MSD to BGP-LS 217
Verifying Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS 217
Feature Information for Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS 218

CHAPTER 23 **RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel** 219
Feature Information for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel 219
Prerequisites for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel 220
Restrictions for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel 220
Information About RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel 221
Benefits of RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel 221
Backup AutoTunnel 221
How to Configure RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel 223
Configuring Explicit Path for Point-to-Point Network Type 223
Configuring Explicit RSVP-TE Tunnel With FRR 224
Verifying RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel 225

CHAPTER 24 ISIS Manual Adjacency SID 227
Feature Information for ISIS Manual Adjacency SID 227
Information About ISIS Manual Adjacency SID 227
Manual Adjacency SID 228
Adjacency SID Advertisement 228
Adjacency SID Forwarding 229
Configuration Prerequisites 229
Configuring Manual Adjacency SID 229
Verifying Manual Adjacency SID 230

CHAPTER 25 OSPF Manual Adjacency SID 231
Feature Information for OSPF Manual Adjacency SID 231
Information About OSPF Manual Adjacency SID 231
Prerequisites for OSPF Manual Adjacency SID 232
Restrictions for OSPF Manual Adjacency SID 232
Manual Adjacency SIDs 232
Manual Adjacency SID Advertisement 233
Manual Adjacency SID Forwarding 233
How to Configure OSPF Manual Adjacency SID 233
Modifying Segment Routing Local Block Range 233
Configuring OSPF Manual Adjacency SID 233
Verifying OSPF Manual Adjacency SID 234

CHAPTER 26 OSPFv2 Segment Routing Strict SPF 237
Feature Information for OSPFv2 Segment Routing Strict SPF 237
Restrictions for OSPFv2 Segment Routing Strict SPF 238
Information About OSPFv2 Segment Routing Strict SPF 238
Why Strict SPF 238
Strict-SPF Capability Advertisement 238
Strict-SPF SID Advertisement in Extended Prefix LSA 239
Interaction with SR-TE and Router Information Base 239
Enabling and Disabling OSPFv2 Segment Routing Strict SPF 239
Configuring OSPFv2 Segment Routing Strict SPF SID 240

Verifying OSPFv2 Segment Routing Strict SPF 240

CHAPTER 27 **Segment Routing OSPFv2 Microloop Avoidance 247**

Feature Information for Segment Routing OSPFv2 Microloop Avoidance 247

Information About Segment Routing OSPFv2 Microloop Avoidance 248

Microloops 248

Preventing Microloops using Segment Routing 251

Prerequisites for Segment Routing OSPFv2 Microloop Avoidance 251

Restrictions for Segment Routing OSPFv2 Microloop Avoidance 252

Configuring Segment Routing OSPFv2 Microloop Avoidance 252

Verifying Segment Routing OSPFv2 Microloop Avoidance 252

CHAPTER 28 **Performance Measurement for Traffic Engineering 253**

Feature Information for Performance Measurement for Traffic Engineering 253

Information about Performance Metrics for Traffic Engineering 254

Overview of Link Delay Measurement 254

Link Delay Metrics for a Computation Interval 255

Link Delay Metrics for Advertisement 255

Global Link Delay Profile 256

Benefits of Link Delay Measurement 257

Restrictions for Link Delay Measurement 258

How to Configure Performance Measurement for Traffic Engineering 258

Configuring Global Link Delay Profile 258

Configuring Link Delay Measurement for an Interface 258

Enabling Monitoring Mode 259

Verifying Link Delay Configuration 260

Viewing Link Delay Information for an Interface 260

Additional Commands 261

Additional References 263

CHAPTER 29 **Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains 265**

Feature Information for Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains 265

Information about Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains 266

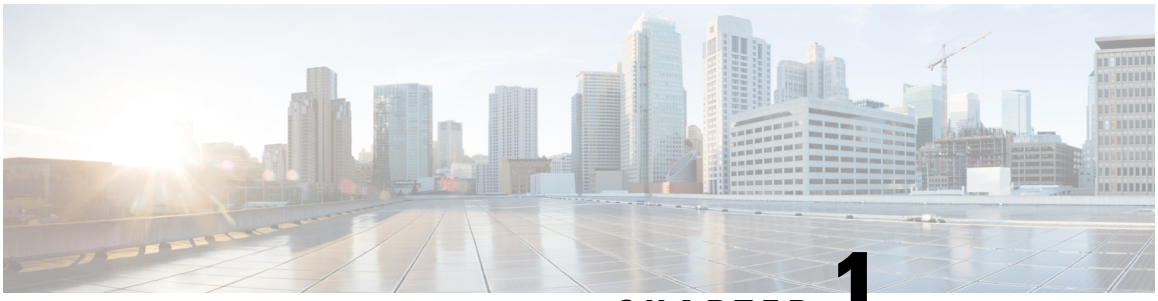
Overview of the Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains 266

How to Configure Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains 266

Configure the Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains 266

Verify the Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains 267

Example: Configure Loopback Prefix SIDs of a BR in Multiple ISIS Domains 268



CHAPTER 1

# Read Me First

### Important Information about Cisco IOS XE 16

Effective Cisco IOS XE Release 3.7.0E for Catalyst Switching and Cisco IOS XE Release 3.17S (for Access and Edge Routing) the two releases evolve (merge) into a single version of converged release—the Cisco IOS XE 16—providing one release covering the extensive range of access and edge products in the Switching and Routing portfolio.

See the *Cisco IOS XE Denali 16.2 Migration Guide for Access and Edge Routers* document for the full list of supported platforms for the various Cisco IOS XE 16 releases and also the migration strategy for the supported products. This document contains key information and steps that will help ensure a successful migration from extant Cisco IOS XE 3.17S releases to the Cisco IOS XE 16.2 release. It also provides key software differences between this release and the Cisco IOS XE Release 3.17S that must be kept in mind during migration. It is critical that you read the information before you begin migration to ensure that you have completed all of the prerequisites and to make sure that you understand the migration process.

### Feature Information

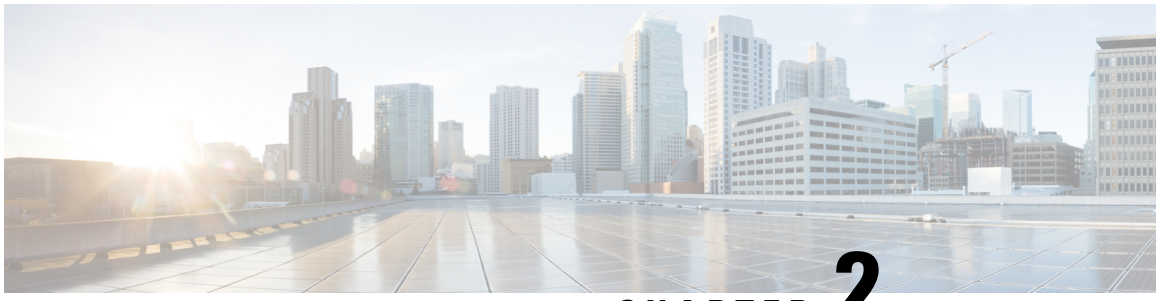
Use Cisco Feature Navigator to find information about feature support, platform support, and Cisco software image support. An account on Cisco.com is not required.

### Related References

- Cisco IOS Command References, All Releases

### Obtaining Documentation and Submitting a Service Request

- 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.



CHAPTER 2

# Introduction to Segment Routing

This chapter introduces the concept of Segment Routing (SR).


- Feature Information for Introduction to Segment Routing, on page 3
- Overview of Segment Routing, on page 3
- How Segment Routing Works, on page 4
- Examples for Segment Routing, on page 5
- Benefits of Segment Routing, on page 6
- Segment Routing Global Block, on page 8
- Additional References for Segment Routing, on page 10


## Feature Information for Introduction to Segment Routing

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

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

***Table 1: Feature Information for Introduction to Segment Routing***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Introduction to Segment Routing | Cisco IOS XE Amsterdam 17.3.2 | Segment Routing (SR) is a flexible, scalable way of doing source routing. |

## Overview of Segment Routing

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

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

- Apply service S

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

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

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

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

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

# How Segment Routing Works

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

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

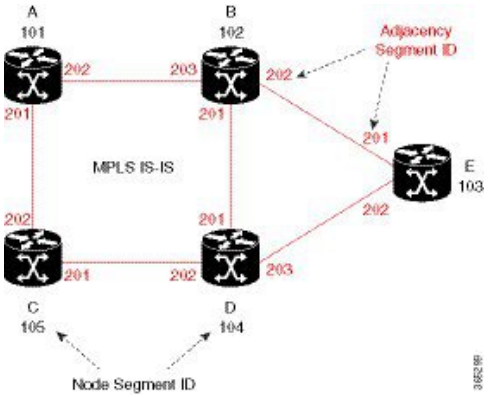
- **Prefix SID**— A segment ID that contains an IP address prefix calculated by an IGP in the service provider core network. Prefix SIDs are globally unique. A prefix segment represents the shortest path (as computed by IGP) to reach a specific prefix; a node segment is a special prefix segment that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

A node segment can be a multi-hop path while an adjacency segment is a one-hop path.

# Examples for Segment Routing

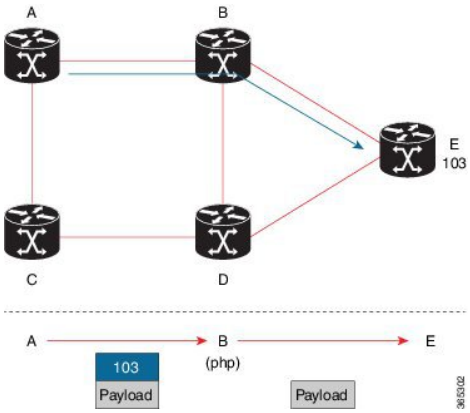
The following figure illustrates an MPLS network with five routers using Segment Routing, IS-IS, a label range of 100 to 199 for node IDs, and 200 and higher for adjacency IDs. IS-IS would distribute IP prefix reachability alongside segment ID (the MPLS label) across the network.

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



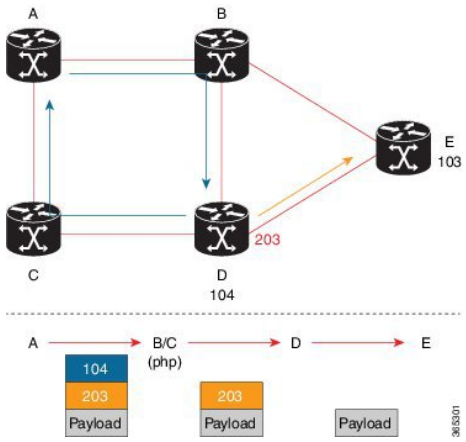

In the previous example, any router sending traffic to router E would push label 103 (router E node segment identifier) to forward traffic using the IS-IS shortest path. The MPLS label-swapping operation at each hop preserves label 103 until the packet arrives at E (Figure 2). On the other hand, adjacency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

***Figure 2: MPLS Label-Swapping Operation***




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

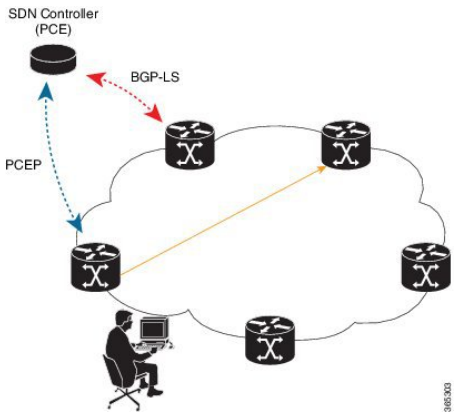
*Figure 3: Router E Destination Path*




# Benefits of Segment Routing

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

  *Figure 4: SDN Controller*

- In addition, segment lists allow complete network virtualization without adding any application state to the network. The state is encoded in the packet as a list of segments. Because the network only maintains

segment state, it can support a large number - and a higher frequency - of transaction-based application requests without creating any burden on the network.

- **Simplified**—
  - When applied to the MPLS data plane, Segment Routing offers the ability to tunnel MPLS services (VPN, VPLS, and VPWS) from an ingress provider edge to an egress provider edge without any other protocol than an IGP (ISIS or OSPF).
  - Simpler operation without separate protocols for label distribution (for example, no LDP or RSVP).
  - No complex LDP or IGP synchronization to troubleshoot.
  - Better utilization of installed infrastructure, for lower capital expenditures (CapEx), with ECMP-aware shortest path forwarding (using node segment IDs).
- **Supports Fast Reroute (FRR)**— Deliver automated FRR for any topology. In case of link or node failures in a network, MPLS uses the FRR mechanism for convergence. With segment routing, the convergence time is sub-50-msec.
- **Large-scale Data Center-**
  - Segment Routing simplifies MPLS-enabled data center designs using Border Gateway Protocol (BGP) RFC 3107 - IPv4 labeled unicast among Top-of-the-Rack/Leaf/Spine switches.
  - BGP distributes the node segment ID, equivalent to IGP node SID.
  - Any node within the topology allocates the same BGP segment for the same switch.
  - The same benefits are provided as for IGP node SID: ECMP and automated FRR (BGP PIC(Prefix Independent Convergence).
  - This is a building block for traffic engineering - SR TE data center fabric optimization.
- **Scalable**—
  - Avoid thousands of labels in LDP database.
  - Avoid thousands of MPLS Traffic Engineering LSP's in the network.
  - Avoid thousands of tunnels to configure.
- **Dual-plane Networks**—
  - Segment Routing provides a simple solution for disjointness enforcement within a so-called “dual-plane” network, where the route to an edge destination from a given plane stays within the plane unless the plane is partitioned.
  - An additional SID “anycast” segment ID allows the expression of macro policies such as: "Flow 1 injected in node A toward node Z must go via plane 1" and "Flow 2 injected in node A towards node Z must go via plane 2."
- **Centralized Traffic Engineering**—
  - Controllers and orchestration platforms can interact with Segment Routing traffic engineering for centralized optimization, such as WAN optimization.

- Network changes such as congestion can trigger an application to optimize (recompute) the placement of segment routing traffic engineering tunnels.
- Segment Routing tunnels are dynamically programmed onto the network from an orchestrator using southbound protocols like PCE.
- Agile network programming is possible since Segment Routing tunnels do not require signaling and per-flow state at midpoints and tail end routers.

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

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

# Segment Routing Global Block

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

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

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

The SRGB label value is calculated as follows:

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

**Restrictions:**

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

# Segment Routing Global Block

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

# Adjacency Segment Identifiers

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

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

# Prefix Segment Identifiers

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

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

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

When a prefix SID value *x* is configured, the prefix SID translates to a label value equivalent to *x* + lower boundary of SRGB. For example, in the platform supporting 1000000 MPLS labels or more if the default SRGB is used, configuring a prefix-SID of 10 for interface Loopback 0 with IPv4 address 1.0.0.1/32 results in assigning the label 9000010 16010 to the prefix 1.0.0.1/32.

**BGP Prefix Segment Identifiers**

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

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

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

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

Each prefix is assigned with its own unique label index.

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

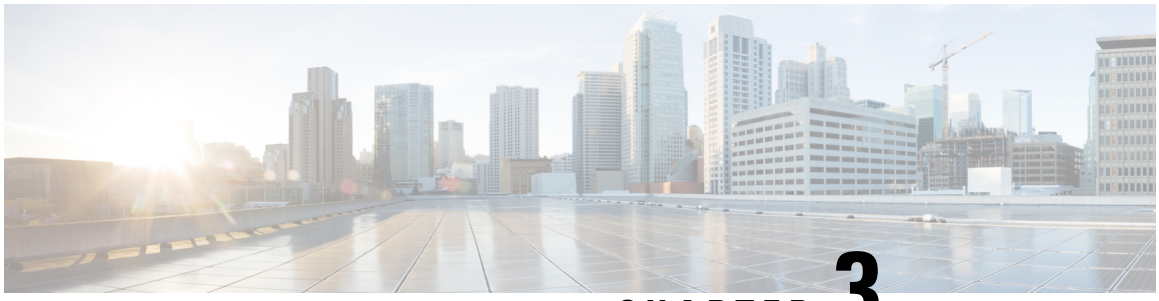
Configure the Segment Routing Global Block (SRGB) in BGP. If the route label path has a label-index attribute and SRGB is configured, then local label route is allocated from SRGB. If label-index is added to redistributed routes using route-policy, then BGP presents label-index as an attribute with the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

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

# Additional References for Segment Routing

### Related Documents

| Related Topic | Document Title |
|---|---|
| Videos | • Introduction to Cisco Segment Routing (YouTube)<br>• Introduction to Cisco Segment Routing (CCO) |



CHAPTER 3

# Segment Routing With IS-IS v4 Node SID

This chapter describes how Segment Routing (SR) works with IS-IS.


- Feature Information for Segment Routing—IS-IS v4 Node SID, on page 11
- Restrictions for Segment Routing With IS-IS v4 Node SID, on page 11
- Information About Segment Routing IS-IS v4 Node SID, on page 12
- How to Configure Segment Routing —IS-IS v4 Node SID, on page 15
- Configuration Examples for Segment Routing —IS-IS v4 Node SID, on page 22
- Additional References for Segment Routing With IS-IS v4 Node SID, on page 23


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

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

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

***Table 2: Feature Information for Segment Routing—IS-IS v4 Node SID***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing—IS-IS v4 Node SID | Cisco IOS XE Amsterdam 17.3.2 | The Segment Routing—ISIS v4 node SID feature provides support for segment routing on IS-IS networks.<br><br>The following commands were introduced or modified: **connected-prefix-sid-map**, **show isis segment-routing**, **isis prefix n-flag-clear**, **explicit-null** |

## Restrictions for Segment Routing With IS-IS v4 Node SID

- Segment routing must be configured at the top level before any routing protocol configuration is allowed under its router configuration sub mode.
- IS-IS protocol SR command is based on per topology (IPv4 address family).
- Effective Cisco IOS-XE Release 3.16, ISIS supports segment routing for IPv4 only.

# Information About Segment Routing IS-IS v4 Node SID

## Segment Routing IS-IS v4 Node SID

Segment Routing relies on a small number of extensions to Cisco Intermediate System-to-Intermediate System (IS-IS) and Open Shortest Path First (OSPF) protocols. There are two levels of configuration required to enable segment routing for a routing protocol instance. The top level segment routing configuration which is managed by segment routing infrastructure component enables segment routing, whereas, segment routing configuration at the router level enables segment routing for a specific address-family of a routing protocol instance. There are three segment routing states:

- SR_NOT_CONFIGURED
- SR_DISABLED
- SR_ENABLED

Segment routing configuration under the IGPs is allowed only if the SR state is either SR_DISABLED or SR_ENABLED. The SR_ENABLED state indicates that there is at least a valid SRGB range reserved through the MFI successfully. You can enable segment routing for IGPs under the router configuration sub mode, through commands. However, IGP segment routing are enabled only after the global SR is configured.

**Note** IS-IS protocol SR command is based on per topology (IPv4 address family).

The SR_ENABLED is a necessary state for any protocol to enable SR, however, it is not a sufficient for enabling SR for a protocol instance. The reason being that the IS-IS still does not have any information about segment routing global block (SRGB) information. When the request to receive information about the SRGB is processed successfully, the IS-IS SR operational state is enabled.

Segment Routing requires each router to advertise its segment routing data-plane capability and the range of MPLS label values that are used for segment routing in the case where global SIDs are allocated. Data-plane capabilities and label ranges are advertised using the SR-capabilities sub-TLV inserted into the IS-IS Router Capability TLV-242 that is defined in RFC4971.

ISIS SR-capabilities sub TLV includes all reserved SRGB ranges. However, the Cisco implementation supports only one SRGB range. The supported IPv4 prefix-SID sub TLV are TLV-135 and TLV-235.

## Prefix-SID Received in Label Switched Path from Remote Routers

Prefix SIDs received in a label switched path (LSP) with a reachability TLV (TLV 135 and 235) are downloaded to the routing information base (RIB) in the same way as BGP downloads per prefix VPN labels, only if the following conditions are met:

- Segment routing is enabled for the topology and address-family.
- Prefix-SID is valid.
- The local label binding to MFI is successful.

**Note**

- For SIDs that do not fit in the specified SID range, labels are not used when updating the RIB. For the cases, where SID fits in the SID range, but does not fit the next-hop neighbor SID range, remote label associated with that path is not installed.
- Node SIDs received in an LSP with reachability TLVs (TLV 135 and 235) are downloaded to RIB only if segment routing is enabled under the corresponding address-family.
- In case of multiple best next hops, if all the next hops do not support segment routing, ISIS treats the instance similar to mismatched labels assigned to the same prefix. That means, IS-IS ignores the labels and installs unlabeled paths for all ECMP paths into the global RIB.

# Segment Routing Adjacency SID Advertisement

Effective with Cisco IOS-XE Release 3.17, IS-IS supports the advertisement of segment routing adjacency SID. An Adjacency Segment Identifier (Adj-SID) represents a router adjacency in Segment Routing.

A segment routing-capable router may allocate an Adj-SID for each of its adjacencies and an Adj-SID sub-TLV is defined to carry this SID in the Adjacency TLVs. IS-IS adjacencies are advertised using one of the IS-Neighbor TLVs below:

- TLV-22 [RFC5305]
- TLV-23 [RFC5311]

IS-IS allocates the adjacency SID for each IS-IS neighbor only if the IS-IS adjacency state is up and IS-IS segment routing internal operational state is enabled. If an adjacency SID allocation failure is due to out-of-label resource, IS-IS retries to allocate the Adj-SID periodically in a default interval (30 seconds).

## Multiple Adjacency-SIDs

Effective with Cisco IOS-XE Release 3.18, multiple adjacency-SIDs are supported. For each protected P2P/LAN adjacency, IS-IS allocates two Adj-SIDs. The backup Adj-SID is only allocated and advertised when FRR (local LFA) is enabled on the interface. If FRR is disabled, then the backup adjacency-SID is released. The persistence of protected adj-SID in forwarding plane is supported. When the primary link is down, IS-IS delays the release of its backup Adj-SID until the delay timer expires. This allows the forwarding plane to continue to forward the traffic through the backup path until the router is converged.

Cisco IOS-XE Release 3.18, IS-IS Adj-SID is changed to be per level based since the forwarding plane is unaware of protocol-specific levels. The allocated and advertised backup Adj-SIDs can be displayed in the output of **show isis neighbor detail** and **show isis data verbose** commands.

# Segment Routing Mapping Server (SRMS)

Segment Routing Mapping Server (SRMS) allows configuration and maintenance of the Prefix-SID mapping policy entries. Effective with Cisco IOS-XE Release 3.17, the IGPs use the active policy of the SRMS to determine the SID values when programming the forwarding plane.

The SRMS provides prefixes to SID/Label mapping policy for the network. IGPs, on the other hand, are responsible for advertising prefixes to SID/Label mapping policy through the Prefix-SID/Label Binding TLV. Active policy information and changes are notified to the IGPs, which use active policy information to update forwarding information.

### Connected Prefix SIDs

Sometimes, a router may install a prefix with a SID that is different than what it advertises to the LSP. For example, if more than one protocol or more than one IGP instance is announcing the same prefix with different SIDs to the SRMS, the SRMS resolves the conflict and announces the winning prefix and SID that may not be the same as the local instance. In that case, the IGP always advertises what it learns from its source LSP although it still tries to install the SID which may be different than what it learns in its LSP. This is done to prevent the IGP from redistributing the SIDs from another protocol or another protocol instance.

## SRGB Range Changes

When IS-IS segment routing is configured, IS-IS must request an interaction with the SRGB before IS-IS SR operational state can be enabled. If no SRGB range is created, IS-IS will not be enabled.

When an SRGB change event occurs, IS-IS makes the corresponding changes in its sub-block entries. IS-IS also advertises the newly created or extended SRGB range in SR-capabilities sub-TLV and updates the prefix-sid sub TLV advertisement.

**Note** In Cisco IOS-XE Release 3.16 only one SRGB range and SRGB extension for the modification are supported.

### SRGB Deletion

When IS-IS receives an SRGB deletion event, it looks for an SRGB entry in the IS-IS SRGB queue list. If an SRGB entry does not exist, IS-IS makes sure that there is no pending SRGB created event. If a pending SRGB creation event is found, then IS-IS removes the SRGB creation event, and completes the SRGB delete processing,

If an SRGB entry is found in the IS-IS SRGB queue, IS-IS locks the SRGB, redistributes the RIBs and un-advertises all prefixed-SIDs that have SID value within the pending delete SRGB range, and un-advertises the SRGB range from SR-capabilities sub TLV. Once IS-IS has completed the SRGB deletion processing, it unlocks the SRGB and deletes the SRGB from its SR sub-block entry.

If there is no valid SRGB after the deletion of the SRGB, IS-IS SR operational state becomes disabled.

## MPLS Forwarding on an Interface

MPLS forwarding must be enabled before segment routing can use an interface. IS-IS is responsible for enabling MPLS forwarding on an interface.

When segment routing is enabled for a IS-IS topology, or IS-IS segment routing operational state is enabled, IS-IS enables MPLS for any interface on which the IS-IS topology is active. Similarly, when segment routing is disabled for a IS-IS topology, IS-IS disables the MPLS forwarding on all interfaces for that topology.

## Segment Routing and LDP Preference

The command **sr-label-preferred** allows the forwarding interface to prefer the segment routing labels over LDP labels for all prefixes in a topology.

## Segment Routing -Traffic Engineering Announcements

IS-IS announces the SR information to TE when it detects that both, IS-IS SR and TE are enabled for at least one level. IS-IS announce only the information that is obtained from the level for which TE is configured.

Similarly, IS-IS instructs TE to delete all announcements when it detects that SR is not enabled or TE is no longer configured on any level.

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

## Configuring Segment Routing

### Before you begin

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

### SUMMARY STEPS

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

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device# enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | segment-routing mpls<br>**Example:**<br>`Device(config-sr)# segment-routing mpls` | Enables the segment feature using the mpls data plane. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | connected-prefix-sid-map<br>**Example:**<br>`Device(config-srmpls)# connected-prefix-sid-map` | Enters a sub-mode where you can configure address-family specific mappings for local prefixes and SIDs. |
| **Step 5** | address-family ipv4<br>**Example:**<br>`Device(config-srmpls-conn)# address-family ipv4` | Specifies IPv4 address prefixes. |
| **Step 6** | 1.1.1.1/32 index 100 range 1<br>**Example:**<br>`Device(config-srmpls-conn-af)# 1.1.1.1/32 100 range 1` | Associates SID 100 with the address 1.1.1.1/32. |
| **Step 7** | exit-address-family<br>**Example:**<br>`Device(config-srmpls-conn-af)# exit-address-family` | Exits the address family. |

# Configuring Segment Routing on IS-IS Network

### Before you begin

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

### SUMMARY STEPS

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

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | router isis<br>**Example:**<br>`Device(config-router)# router isis` | Enables the IS-IS routing protocol and enters router configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | net network-entity-title<br>**Example:**<br>`Device(config-router)# net 49.0000.0000.0003.00` | Configures network entity titles (NETs) for the routing instance. |
| **Step 3** | metric-style wide<br>**Example:**<br>`Device(config-router)# metric-style wide` | Configures the device to generate and accept only wide link metrics. |
| **Step 4** | **segment-routing** mpls<br>**Example:**<br>`Device(config-router)# segment-routing mpls` | Configures segment routing operation state. |
| **Step 5** | exit<br>**Example:**<br>`Device(config-router)# exit` | Exits segment routing mode and returns to the configuration terminal mode. |
| **Step 6** | show isis segment-routing<br>**Example:**<br>`Device# show is-is segment-routing` | Displays the current state of the IS-IS segment routing. |

### Example

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

```
Device# show isis segment-routing

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

# Configuring Prefix-SID for IS-IS

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

### Before you begin

Segment routing must be enabled on the corresponding address family.

### SUMMARY STEPS

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

### DETAILED STEPS

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

# Configuring Prefix Attribute N-flag-clear

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

### SUMMARY STEPS

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

### DETAILED STEPS

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

# Configuring Explicit Null Attribute

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

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

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. segment-routing mpls

4. set-attributes
5. address-family ipv4
6. explicit-null
7. exit-address-family

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device# enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | segment-routing mpls<br>**Example:**<br>`Device(config)# segment-routing mpls` | Configures segment routing mpls mode. |
| **Step 4** | set-attributes<br>**Example:**<br>`Device(config-srmpls)# set-attributes` | Sets the attribute. |
| **Step 5** | address-family ipv4<br>**Example:**<br>`Device(config-srmpls-attr)# address-family ipv4` | Specifies the IPv4 address family and enters router address family configuration mode. |
| **Step 6** | explicit-null<br>**Example:**<br>`Device(config-srmpls-attr-af)# explicit-null` | Specifies the explicit-null. |
| **Step 7** | exit-address-family<br>**Example:**<br>`Device(config-srmpls-attr-af)# exit-address-family` | Exits the address family. |

# Configuring Segment Routing Label Distribution Protocol Preference

### SUMMARY STEPS

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

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device# enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | segment-routing mpls<br>**Example:**<br>`Device(config)# segment-routing mpls` | Configures segment routing mpls mode. |
| **Step 4** | set-attributes<br>**Example:**<br>`Device(config-srmpls)# set-attributes` | Sets the attribute. |
| **Step 5** | address-family ipv4<br>**Example:**<br>`Device(config-srmpls-attr)# address-family ipv4` | Specifies the IPv4 address family and enters router address family configuration mode. |
| **Step 6** | sr-label-preferred<br>**Example:**<br>`Device(config-srmpls-attr-af)# sr-label-preferred` | Specifies SR label to be preferred over the LDP. |
| **Step 7** | exit-address-family<br>**Example:** | Exits the address family. |

| | Command or Action | Purpose |
|---|---|---|
| | `Device(config-srmpls-attr-af)# exit-address-family` | |

# Configuring IS-IS SRMS

The following command enables the IS-IS SRMS and allows IS-IS to advertise local mapping entries. IS-IS does not send remote entries to the SRMS library. However, IS-IS uses the SRMS active policy, which is computed based only on the locally configured mapping entries.

```
[no] segment-routing prefix-sid-map advertise-local
```

# Configuring IS-IS SRMS Client

By default, the IS-IS SRMS client mode is enabled. IS-IS always sends remote prefix-sid-mapping entries received through LSP to SRMS. The SRMS active policy is calculated based on local and remote mapping entries.

The following command disables the prefix-sid-mapping client functionality and it is configured on the receiver side.

```
segment-routing prefix-sid-map receive [disable]
```

# Configuring IS-IS SID Binding TLV Domain Flooding

By default, the IS-IS SRMS server does not flood SID binding entries within the routing domain. From Cisco IOS-XE Release 3.18, the optional keyword **domain-wide** is added in the IS-IS SRMS server mode command to enable the SID and Label binding TLV flooding functionality.

```
segment-routing prefix-sid-map advertise-local [domain-wide]
```

The **domain-wide** keyword enables the IS-IS SRMS server to advertise SID binding TLV across the entire routing domain.

**Note** The option is valid only if IS-IS SRMS performs in the SRMS server mode.

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

## Example: Configuring Segment Routing on IS-IS Network

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

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

## Example: Configuring Explicit Null Attribute

The following is an example for configuring explicit null attribute:

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

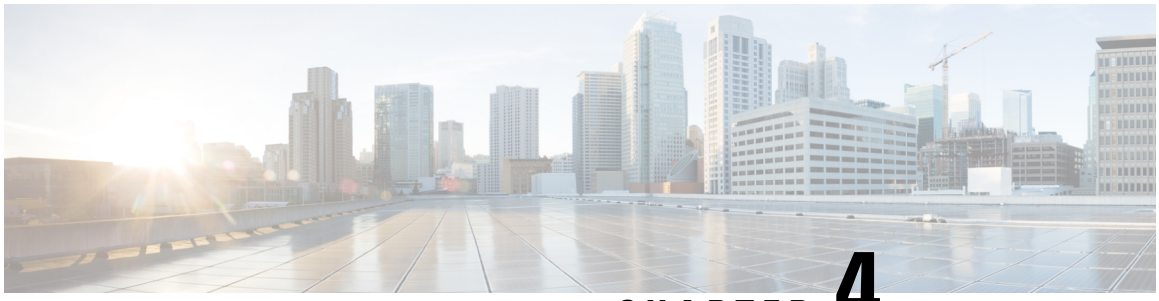
# Additional References for Segment Routing With IS-IS v4 Node SID

### Related Documents

| Related Topic | Document Title |
|---|---|
| IP Routing ISIS commands | Cisco IOS IP Routing ISIS commands |

### RFCs

| RFC | Title |
|---|---|
| RFC4971 | Intermediate System to Intermediate System (IS-IS) Extensions for Advertising Router Information |
| RFC5305 | IS-IS Extensions for Traffic Engineering. Defines the advertisement of router IDs for IPv4. |
| RFC6119 | IPv6 Traffic Engineering in IS-IS. Defines the advertisement of router IDs for IPv6. |



CHAPTER 4

# IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute

This document describes the functionalities and IS-IS implementation of IP Fast Re-Route feature (IPFRR) using Segment Routing (SR) Topology Independent Loop Free Alternative (TI-LFA) link protection.


- Feature Information for IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute, on page 25
- Prerequisites for IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute, on page 26
- Information About IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute, on page 27
- How to Configure IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute, on page 29


## Feature Information for IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute

*Table 3: Feature Information for IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute*

| Feature Name | Releases | Feature Information |
|---|---|---|
| IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute | Cisco IOS XE Amsterdam 17.3.2 | The following commands were introduced or modified:<br>**fast-reroute ti-lfa {level-1 \| level-2} [maximum-metric value]** , isis fast-reroute ti-lfa protection level-1 disable, **isis fast-reroute ti-lfa protection {level-1 \| level-2} [maximum-metric value]**, **show running all \| section interface interface-name**, **show running all \| section router isis**. |

# Prerequisites for IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute

- Enable TI-LFA on all the nodes, before configuring SR-TE for TI-LFA.

```
mpls traffic-eng tunnels
!
segment-routing mpls
 connected-prefix-sid-map
  address-family ipv4
   1.1.1.1/32 index 11 range 1
  exit-address-family
 !
interface Loopback1
 ip address 1.1.1.1 255.255.255.255
 ip router isis 1
!
interface Tunnel1
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng path-option 10 explicit name IP_PATH segment-routing
!
interface GigabitEthernet2
 ip address 192.168.1.1 255.255.255.0
 ip router isis 1
 negotiation auto
 mpls traffic-eng tunnels
 isis network point-to-point
!
interface GigabitEthernet3
 ip address 192.168.2.1 255.255.255.0
 ip router isis 1
 negotiation auto
 mpls traffic-eng tunnels
 isis network point-to-point
!
router isis 1
 net 49.0001.0010.0100.1001.00
 is-type level-1
 metric-style wide
 log-adjacency-changes
 segment-routing mpls
 fast-reroute per-prefix level-1 all
 fast-reroute ti-lfa level-1
 mpls traffic-eng router-id Loopback1
 mpls traffic-eng level-1
!
ip explicit-path name IP_PATH enable
 next-address 4.4.4.4
 next-address 5.5.5.5
 next-address 6.6.6.6
```

- If a microloop gets created between routers in case of primary and secondary path switch over you need to bring down the convergence time. Use the **microloop avoidance rib-update-delay** command to bring down the convergence time:

```
router isis ipfrr
net 49.0001.0120.1201.2012.00
is-type level-2-only
metric-style wide
log-adjacency-changes
segment-routing mpls
segment-routing prefix-sid-map advertise-local
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
microloop avoidance rib-update-delay 10000
```

- Enable MPLS-TE nonstop routing (NSR) and IS-IS nonstop forwarding (NSF) to reduce or minimize traffic loss after a high availability (HA) switch over. Use the **mpls traffic-eng nsr** command in global exec mode.

```
mpls traffic-eng nsr
```

Use the **nsf** command under IS-IS.

```
router isis
nsf cisco
nsf interval 0
```

# Information About IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute

When the local LFA and remote LFA are enabled, there is a good coverage of the prefixes to be protected. However, for some rare topologies that do not have a PQ intersect node, both local and remote LFA will fail to find a release node to protect the failed link. Furthermore, there is no way to prefer a post-convergence path, as the two algorithms have no knowledge of the post-convergence characteristics of the LFA.

To overcome the above limitation, effective Cisco IOS-XE Release 3.18, topology-independent LFA (TI-LFA) is supported on an SR-enabled network.

## Topology-Independent Loop Free Alternate

TI-LFA provides supports for the following:

- Link Protection—The LFA provides repair path for failure of the link.
- Local LFA—Whenever a local LFA on the post convergence path is available, it is preferred over TI-LFA because local LFA does not require additional SID for the repair path. That is, the label for the PQ node is not needed for the release node.
- Local LFA for extended P space—For nodes in the extended P space, local LFA is still the most economical method for the repair path. In this case, TI-LFA will not be chosen.
- Tunnel to PQ intersect node—This is similar to remote LFA except that the repair path is guaranteed on the post convergence path using TI-LFA.
- Tunnel to PQ disjoint node—This capability is unique to the TI-LFA in the case when local and remote LFA cannot find a repair path.
- Tunnel to traverse multiple intersect or disjoint PQ nodes, up to the platform's maximum supported labels—TI-LFA provides complete coverage of all prefixes.

- P2P interfaces for the protected link—TI-LFA protects P2P interfaces.
- Asymmetrical links—The ISIS metrics between the neighbors are not the same.
- Multi-homed (anycast) prefix protection—The same prefix may be originated by multiple nodes.
- Protected prefix filtering—The route-map includes or excludes a list of prefixes to be protected and the option to limit the maximum repair distance to the release node.
- Tiebreakers—A subset of existing tiebreakers, applicable to TI-LFA, is supported.

## Topology Independent Loop Free Alternate Tie-break

Local and remote LFA use default or user-configured heuristics to break the tie when there is more than one path to protect the prefix. The attributes are used to trim down the number of repair paths at the end of the TI-LFA link protection computation before the load balancing. Local LFA and remote LFA support the following tiebreakers:

- Linecard-disjoint—Prefers the line card disjoint repair path
- Lowest-backup-path-metric—Prefers the repair path with lowest total metric
- Node-protecting—Prefers node protecting repair path
- SRLG-disjoint—Prefers SRLG disjoint repair path
- Load-sharing—Distributes repair paths equally among links and prefixes

When there are two repair paths for a particular prefix, the path that the output port on different line card than that of the primary port is chosen as the repair path. For TI-LFA link protection, the following tiebreakers are supported:

- Linecard-disjoint—Prefers the line card disjoint repair path.
- LC disjoint index—If both the repair paths are on the same line card as that of the primary path, then, both paths are considered as candidates. If one of the path is on a different line card, then that path is chosen as the repair path.
- SRLG index—If both the repair paths have the same SRLG ID as that of the primary path, then, both the paths are considered as candidates. If one of the path has a different srlg id, then path is chosen as the repair path.
- Node-protecting—For TI-LFA node protection, the protected node is removed when computing the post-convergence shortest path. The repair path must direct traffic around the protected node.

The SRLG ID can be configured for each interface. When there are two repair paths for a prefix, the configured SRLG ID for the repair path is compared with that of the primary path SRLG ID. If the SRLG IDs for the secondary path is different than that of the primary, that path is chosen as the repair path. This policy comes into effect only when the primary path is configured with an SRLG ID. It is possible to configure both node and SRLG protection modes for the same interface or the same protocol instance. In that case, an additional TI-LFA node-SRLG combination protection algorithm is run. The TI-LFA node-SRLG combination algorithm removes the protected node and all members of the interface with the same SRLG group when computing the post-convergence SPT.

## Interface Fast Reroute Tiebreakers

Interface fast reroute (FRR) tiebreakers are also needed for TI-LFA node and SRLG protection. When interface and protocol instance FRR tiebreakers both are configured, the interface FRR tiebreakers take precedence over the protocol instance. When interface FRR tiebreakers are not configured, the interface inherits the protocol instance FRR tiebreakers.

The following interface FRR tiebreaker commands apply only to the particular interface.

```
isis fast-reroute tie-break
[level-1 | level-2] linecard-disjoint
priority
isis fast-reroute tie-break
[level-1 | level-2] lowest-backup-metric
priority
isis fast-reroute tie-break
[level-1 | level-2] node-protecting
priority
isis fast-reroute tie-break
[level-1 | level-2] srlg-disjoint
priority
isis fast-reroute tie-break
[level-1 | level-2] default
```

Tie-breaker default and explicit tie-breaker on the same interface are mutually exclusive.

The following tie-breakers are enabled by default on all LFAs:

- linecard-disjoint
- lowest-backup-metric
- srlg-disjoint

Effective with Cisco IOS-XE Release 3.18, node-protecting tie-breaker is disabled by default.

# How to Configure IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute

Perform the following steps to configure Link-protection Topology Independent Loop Free Alternate Fast Reroute.

## Configuring Topology Independent Loop Free Alternate Fast Reroute

You can enable TI-LFA using any of the following two methods:

1. **Protocol enablement**—Enables TI-LFA in router isis mode for all IS-IS interfaces. Optionally, use the interface command to exclude the interfaces on which TI-LFA should be disabled.

   For example, to enable TI-LFA for all IS-IS interfaces:

   ```
   router isis 1
   fast-reroute per-prefix {level-1 | level-2}
   fast-reroute ti-lfa {level-1 | level-2} [maximum-metric value]
   ```

**Note** The **isis fast-reroute protection level-x** command enables local LFA and is required to enable TI-LFA.

2. **Interface enablement**—Enable TI-LFA selectively on each interface.

   ```
   interface interface-name
   isis fast-reroute protection {level-1 | level-2}
   isis fast-reroute ti-lfa protection {level-1 | level-2} [maximum-metric value]
   ```

The **maximum-metric** option specifies the maximum repair distance which a node is still considered eligible as a release node.

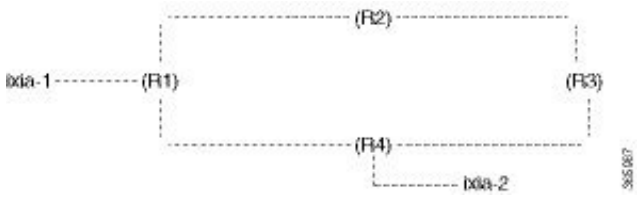
When both interface and protocol are TI-LFA enabled, the interface configuration takes precedence over the protocol configuration. TI-LFA is disabled by default.

To disable TI-LFA on a particular interface, use the following command:

```
interface interface-name
isis fast-reroute ti-lfa protection level-1 disable
```

# Configuring Topology Independent Loop Free Alternate With Mapping Server

Consider the following topology to understand the configuration:




- IXIA-2 injects ISIS prefixes, and IXIA-1 sends one-way traffic to IXIA-2

  .
- In R1 10,000 prefixes are configured in the segment-routing mapping-server.

The configuration on R1 is:

```
configure terminal
segment-routing mpls
global-block 16 20016
!
connected-prefix-sid-map
address-family ipv4
11.11.11.11/32 index 11 range 1
exit-address-family
!
!
mapping-server
!
prefix-sid-map
address-family ipv4
120.0.0.0/24 index 2 range 1 attach
200.0.0.0/24 index 1 range 1 attach
192.168.0.0/24 index 100 range 10000 attach
exit-address-family
!
!
!
!
interface Loopback0
ip address 11.11.11.11 255.255.255.255
ip router isis ipfrr
!
interface GigabitEthernet0/1/0
ip address 14.0.0.1 255.255.255.0
ip router isis ipfrr
```

```
negotiation auto
isis network point-to-point
!
interface GigabitEthernet0/1/2
ip address 11.0.0.1 255.255.255.0
ip router isis ipfrr
negotiation auto
isis network point-to-point
!
interface GigabitEthernet0/1/4
ip address 200.0.0.1 255.255.255.0
ip router isis ipfrr
negotiation auto
isis network point-to-point
!
router isis ipfrr
net 49.0001.0110.1101.1011.00
is-type level-2-only
metric-style wide
log-adjacency-changes
nsf cisco
segment-routing mpls
segment-routing prefix-sid-map advertise-local
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
microloop avoidance rib-update-delay 10000
```

On R2 the configuration is

```
configure terminal
!
!
segment-routing mpls
!
connected-prefix-sid-map
address-family ipv4
12.12.12.12/32 index 12 range 1
exit-address-family
!
!
interface Loopback0
ip address 12.12.12.12 255.255.255.255
ip router isis ipfrr
!
interface GigabitEthernet0/1/0
ip address 12.0.0.1 255.255.255.0
ip router isis ipfrr
negotiation auto
isis network point-to-point
!
interface GigabitEthernet0/1/1
ip address 11.0.0.2 255.255.255.0
ip router isis ipfrr
negotiation auto
isis network point-to-point
!
router isis ipfrr
net 49.0001.0120.1201.2012.00
is-type level-2-only
metric-style wide
log-adjacency-changes
nsf cisco
segment-routing mpls
segment-routing prefix-sid-map advertise-local
```

```
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
microloop avoidance rib-update-delay 10000
!
```

On R3 the configuration is

```
configure terminal
!
mpls traffic-eng tunnels
!
segment-routing mpls
!
connected-prefix-sid-map
address-family ipv4
13.13.13.13/32 index 13 range 1
exit-address-family
!
!
interface Loopback0
ip address 13.13.13.13 255.255.255.255
ip router isis ipfrr
!
interface GigabitEthernet0/0/4
ip address 13.0.0.1 255.255.255.0
ip router isis ipfrr
load-interval 30
speed 1000
no negotiation auto
isis network point-to-point
!
interface GigabitEthernet0/0/5
ip address 12.0.0.2 255.255.255.0
ip router isis ipfrr
negotiation auto
isis network point-to-point
!
router isis ipfrr
net 49.0001.0130.1301.3013.00
is-type level-2-only
metric-style wide
log-adjacency-changes
nsf cisco
segment-routing mpls
segment-routing prefix-sid-map advertise-local
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
microloop avoidance rib-update-delay 10000
!
```

On R4 the configuration is:

```
configure terminal
!
mpls traffic-eng tunnels
!
segment-routing mpls
!
connected-prefix-sid-map
address-family ipv4
14.14.14.14/32 index 14 range 1
exit-address-family
!
```

```
!
interface Loopback0
ip address 14.14.14.14 255.255.255.255
ip router isis ipfrr
!
interface GigabitEthernet0/0/0
ip address 14.0.0.2 255.255.255.0
ip router isis ipfrr
negotiation auto
isis network point-to-point
!
interface GigabitEthernet0/0/3
ip address 13.0.0.2 255.255.255.0
ip router isis ipfrr
speed 1000
no negotiation auto
isis network point-to-point
!
interface GigabitEthernet0/0/5
ip address 120.0.0.1 255.255.255.0
ip router isis ipfrr
speed 1000
no negotiation auto
isis network point-to-point
!
router isis ipfrr
net 49.0001.0140.1401.4014.00
is-type level-2-only
metric-style wide
log-adjacency-changes
nsf cisco
segment-routing mpls
segment-routing prefix-sid-map advertise-local
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
microloop avoidance rib-update-delay 10000
!
```

# Examples: Configuring IS-IS Link-protection Topology Independent Loop Free Alternate Fast Reroute

Example 1: In the following example, local LFA is configured with linecard-disjoint and srlg-disjoint tiebreakers. Linecard-disjoint is given preference with a lower priority value (10) than the srlg-disjoint (11).

```
router isis access
 net 49.0001.2037.0685.b002.00
 metric-style wide
 fast-flood 10
 max-lsp-lifetime 65535
 lsp-refresh-interval 65000
 spf-interval 5 50 200
 prc-interval 5 50 200
 lsp-gen-interval 5 5 200
 log-adjacency-changes
 nsf ietf
 segment-routing mpls
 fast-reroute per-prefix level-1 all – configures the local LFA
 fast-reroute per-prefix level-2 all
 fast-reroute remote-lfa level-1 mpls-ldp - enables rLFA (optional)
 fast-reroute remote-lfa level-2 mpls-ldp
```

```
 fast-reroute ti-lfa level-1 - enables TI-LFA
 microloop avoidance rib-update-delay 10000
 bfd all-interfaces
```

Example 2—Enable TI-LFA node-protecting tie-breaker on all ISIS level-2 interfaces with priority 100. All other tiebreakers are disabled.

```
router isis
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
fast-reroute tie-break level-2 node-protecting 100
```

Example 3—Enable TI-LFA node-protecting tie-breaker with priority 100 and TI-LFA SRLG protection with priority 200 on all IS-IS level-2 interfaces. All other tiebreakers are disabled because the node-protecting tie-breaker is configured.

```
router isis
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
fast-reroute tie-break level-2 node-protecting 100
fast-reroute tie-break level-2 srlg-disjoint 200
```

Example 4—Enable TI-LFA node-protecting tie-breaker with priority 100 on all ISIS level-2 interfaces except on Ethernet0/0. For those IS-IS interfaces, all other tiebreakers are disabled. Ethernet0/0 overwrites the inheritance and uses the default set of tiebreakers with linecard-disjoint, lowest-backup-path-metric, srlg-disjoint enabled.

```
router isis
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
fast-reroute tie-break level-2 node-protecting 100
!
interface ethernet0/0
ip router isis
isis fast-reroute tie-break level-2 default
```

Example 5—Enable TI-LFA using the default tiebreaker on all IS-IS interfaces except on Ethernet0/0. On Ethernet0/0 enable TI-LFA node-protecting with priority 100 and disable all other tiebreakers.

```
router isis
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
!
interface ethernet0/0
ip router isis
isis fast-reroute tie-break level-2 node-protecting 100
```

Example 6—Enable TI-LFA node-protecting tie-breaker with priority 200 and linecard-disjoint tie-breaker with priority 100 on all ISIS level-2 interfaces. All other tiebreakers are disabled.

```
router isis
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
fast-reroute tie-break level-2 linecard-disjoint 100
fast-reroute tie-break level-2 node-protecting 200
```

# Verifying the Tiebreaker

To view tiebreakers enabled on the interface, use the following command:

**show running all | section interface** *interface-name*

To view tiebreakers enabled on the router mode, use the following command:

**show running all | section router isis**

# Verifying the Primary and Repair Paths

In this example, 1.1.1.1 is the protecting neighbor and 4.4.4.4 is the neighbor on the protecting link.

```
Router#
show ip cef 1.1.1.1
1.1.1.1/32
  nexthop 1.1.1.1 GigabitEthernet0/2/0 label [explicit-null|explicit-null]() - slot 2 is
primary interface
    repair: attached-nexthop 24.0.0.2 TenGigabitEthernet0/3/0 - slot 3 is repair interface
  nexthop 24.0.0.2 TenGigabitEthernet0/3/0 label [explicit-null|explicit-null]()
    repair: attached-nexthop 1.1.1.1 GigabitEthernet0/2/0
Router#
show ip cef 4.4.4.4
4.4.4.4/32
  nexthop 4.4.4.4 GigabitEthernet0/2/3 label [explicit-null|16004]() - slot 2 is primary
interface
    repair: attached-nexthop 5.5.5.5 MPLS-SR-Tunnel2
Router# show ip cef 4.4.4.4 int
4.4.4.4/32, epoch 3, RIB[I], refcnt 6, per-destination sharing
  sources: RIB, Adj, LTE
  feature space:
    IPRM: 0x00028000
    Broker: linked, distributed at 4th priority
    LFD: 4.4.4.4/32 2 local labels
    dflt local label info: global/877 [0x3]
    sr local label info: global/16004 [0x1B]
        contains path extension list
        dflt disposition chain 0x46654200
          label implicit-null
          FRR Primary
            <primary: IP adj out of GigabitEthernet0/2/3, addr 4.4.4.4>
        dflt label switch chain 0x46654268
          label implicit-null
          TAG adj out of GigabitEthernet0/2/3, addr 4.4.4.4
        sr disposition chain 0x46654880
          label explicit-null
          FRR Primary
            <primary: TAG adj out of GigabitEthernet0/2/3, addr 4.4.4.4>
        sr label switch chain 0x46654880
          label explicit-null
          FRR Primary
            <primary: TAG adj out of GigabitEthernet0/2/3, addr 4.4.4.4>
  subblocks:
    Adj source: IP adj out of GigabitEthernet0/2/3, addr 4.4.4.4 464C6620
      Dependent covered prefix type adjfib, cover 0.0.0.0/0
  ifnums:
    GigabitEthernet0/2/3(11): 4.4.4.4
    MPLS-SR-Tunnel2(1022)
  path list 3B1FC930, 15 locks, per-destination, flags 0x4D [shble, hvsh, rif, hwcn]
    path 3C04D5E0, share 1/1, type attached nexthop, for IPv4, flags [has-rpr]
```

```
     MPLS short path extensions: [rib | lblmrg | srlbl] MOI flags = 0x21 label explicit-null

      nexthop 4.4.4.4 GigabitEthernet0/2/3 label [explicit-null|16004](), IP adj out of
GigabitEthernet0/2/3, addr 4.4.4.4 464C6620
        repair: attached-nexthop 5.5.5.5 MPLS-SR-Tunnel2 (3C04D6B0)
    path 3C04D6B0, share 1/1, type attached nexthop, for IPv4, flags [rpr, rpr-only]
      MPLS short path extensions: [rib | lblmrg | srlbl] MOI flags = 0x1 label 16004
      nexthop 5.5.5.5 MPLS-SR-Tunnel2 label 16004(), repair, IP midchain out of
MPLS-SR-Tunnel2 46CE2440
  output chain:
    label [explicit-null|16004]()
    FRR Primary (0x3B209220)
      <primary: TAG adj out of GigabitEthernet0/2/3, addr 4.4.4.4 464C6480> - primary path
      <repair:  TAG midchain out of MPLS-SR-Tunnel2 46CE22A0
                label 16()
                label 16003()
                TAG adj out of TenGigabitEthernet0/3/0, addr 24.0.0.2 46CE25E0> - repair
path
```

# Verifying the IS-IS Segment Routing Configuration

```
Router# show isis segment-routing
 ISIS protocol is registered with MFI
 ISIS MFI Client ID:0x63
 Tag Null - Segment-Routing:
   SR State:SR_ENABLED
   Number of SRGB:1
   SRGB Start:14000, Range:1001, srgb_handle:0xE0934788, srgb_state: created
   Address-family IPv4 unicast SR is configured
     Operational state: Enabled
```

The command with keyword **global-block** displays the SRGB and the range for LSPs.

```
Router# show isis segment-routing global-block
IS-IS Level-1 Segment-routing Global Blocks:
System ID             SRGB Base    SRGB Range
nevada                20000        4001
arizona             * 16000        1000
utah                  40000        8000
```

The **show isis segment-routing prefix-sid-map** command with keyword **advertise** displays the prefix-sid maps that the router advertises.

```
Roouter# show isis segment-routing prefix-sid-map adv
IS-IS Level-1 advertise prefix-sid maps:
Prefix               SID Index    Range        Flags
16.16.16.16/32       101          1
16.16.16.17/32       102          1            Attached
```

The **show isis segment-routing prefix-sid-map** command with keyword **receive** displays the prefix-sid maps that the router receives.

```
Router #sh isis segment-routing prefix-sid-map receive
IS-IS Level-1 receive prefix-sid maps:
Host                 Prefix               SID Index    Range        Flags
utah                 16.16.16.16/32       101          1
                     16.16.16.17/32       102          1            Attached
```

To display the connected-SIDs found in the LSPs and passed to the mapping server component, use the **show isis segment-routing connected-sid** command.

```
Router# show isis segment-routing connected-sid
IS-IS Level-1 connected-sids
Host                  Prefix              SID Index    Range        Flags
nevada              * 1.1.1.2/32          1002         1
                      2.2.2.2/32          20           1
                      100.1.1.10/32       10           1
colorado              1.1.1.3/32          33           1
                      1.1.1.6/32          6            1
IS-IS Level-2 connected-sids
Host                 Prefix               SID Index    Range        Flags
```

# Verifying the IS-IS Topology Independent Loop Free Alternate Tunnels

```
Router# show isis fast-reroute ti-lfa tunnel
Fast-Reroute TI-LFA Tunnels:
Tunnel  Interface  Next Hop         End Point        Label     End Point Host
MP1     Et1/0      30.1.1.4         1.1.1.2          41002     nevada
MP2     Et0/0      19.1.1.6         1.1.1.6          60006     colorado
                                    1.1.1.2          16        nevada
MP3     Et0/0      19.1.1.6         1.1.1.6          60006     colorado
                                    1.1.1.2          16        nevada
                                    1.1.1.5          70005     wyoming
```

# Verifying the Segment Routing Traffic Engineering With Topology Independent Loop Free Alternate Configuration

```
Router# show mpls traffic-eng tunnels tunnel1
Name: PE1                              (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type explicit IP_PATH (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 0 [0] bw-based
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 4 hours, 25 minutes
      Time since path change: 4 hours, 21 minutes
      Number of LSP IDs (Tun_Instances) used: 37
    Current LSP: [ID: 37]
      Uptime: 4 hours, 21 minutes
  Tun_Instance: 37
  Segment-Routing Path Info (isis  level-1)
    Segment0[Node]: 4.4.4.4, Label: 16014
    Segment1[Node]: 5.5.5.5, Label: 16015
```

```
    Segment2[Node]: 6.6.6.6, Label: 16016
Router# show isis fast-reroute ti-lfa tunnel

Tag 1:
Fast-Reroute TI-LFA Tunnels:
Tunnel  Interface  Next Hop         End Point        Label     End Point Host
MP1     Gi2        192.168.1.2      6.6.6.6          16016     SR_R6
MP2     Gi3        192.168.2.2      6.6.6.6          16016     SR_R6
Router# show frr-manager client client-name ISIS interfaces detail
TunnelI/F : MP1
  Type : SR
  Next-hop : 192.168.1.2
  End-point : 6.6.6.6
  OutI/F : Gi2
  Adjacency State : 1
  Prefix0 : 6.6.6.6(Label : 16016)
TunnelI/F : MP2
  Type : SR
  Next-hop : 192.168.2.2
  End-point : 6.6.6.6
  OutI/F : Gi3
  Adjacency State : 1
  Prefix0 : 6.6.6.6(Label : 16016)
Router# show ip cef 6.6.6.6 internal

6.6.6.6/32, epoch 2, RIB[I], refcnt 6, per-destination sharing
  sources: RIB, LTE
  feature space:
    IPRM: 0x00028000
    Broker: linked, distributed at 1st priority
    LFD: 6.6.6.6/32 1 local label
    sr local label info: global/16016 [0x1A]
        contains path extension list
        sr disposition chain 0x7FC6B0BF2AF0
          label implicit-null
          IP midchain out of Tunnel1
          label 16016
          FRR Primary
            <primary: label 16015
                      TAG adj out of GigabitEthernet3, addr 192.168.2.2>
        sr label switch chain 0x7FC6B0BF2B88
          label implicit-null
          TAG midchain out of Tunnel1
          label 16016
          FRR Primary
            <primary: label 16015
                      TAG adj out of GigabitEthernet3, addr 192.168.2.2>
  ifnums:
    Tunnel1(13)
  path list 7FC6B0BBDDE0, 3 locks, per-destination, flags 0x49 [shble, rif, hwcn]
    path 7FC7144D4300, share 1/1, type attached nexthop, for IPv4
      MPLS short path extensions: [rib | prfmfi | lblmrg | srlbl] MOI flags = 0x3 label
implicit-null
      nexthop 6.6.6.6 Tunnel1, IP midchain out of Tunnel1 7FC6B0BBB440
  output chain:
    IP midchain out of Tunnel1 7FC6B0BBB440
    label [16016|16016]
    FRR Primary (0x7FC714515460)
      <primary: label 16015
                TAG adj out of GigabitEthernet3, addr 192.168.2.2 7FC6B0BBB630>
      <repair:  label 16015
                label 16014
                TAG midchain out of MPLS-SR-Tunnel1 7FC6B0BBAA90
```

```
label 16016
TAG adj out of GigabitEthernet2, addr 192.168.1.2 7FC6B0BBBA10>
```

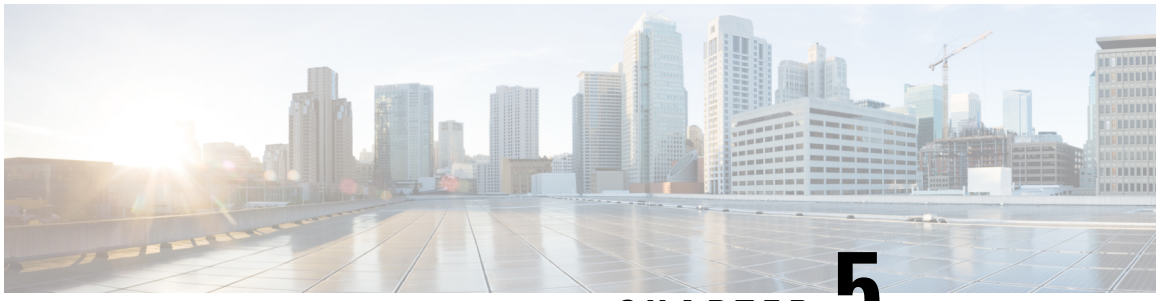
**Note** To ensure a less than 50 msec traffic protection with TI-LFA, SR-TE with dynamic path option must use the backup adjacency SID.

To create an SR-TE with dynamic path option, use the following configuration on every router in the topology:

```
router isis 1
fast-reroute per-prefix level-1 all
```

At the tunnel head-end router:

```
interface Tunnel1
ip unnumbered Loopback1
tunnel mode mpls traffic-eng
tunnel destination 6.6.6.6
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng path-option 1 dynamic segment-routing
tunnel mpls traffic-eng path-selection segment-routing adjacency protected
```



CHAPTER 5

# Segment Routing Traffic Engineering With IS-IS

This chapter describes how segment routing traffic engineering (SR-TE) can be implemented using IS-IS.


- Feature Information for Segment Routing -Traffic Engineering With IS-IS, on page 41
- Restrictions for Segment Routing-Traffic Engineering With IS-IS, on page 41
- Information About Segment Routing Traffic Engineering With IS-IS, on page 42
- How to Configure Segment Routing Traffic Engineering With IS-IS, on page 48


## Feature Information for Segment Routing -Traffic Engineering With IS-IS

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

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

***Table 4: Feature Information for Segment Routing -Traffic Engineering With IS-IS***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing-Traffic Engineering With IS-IS | Cisco IOS XE Amsterdam 17.3.2 | The following commands were introduced or modified:<br>**mpls traffic-eng nsr**, **show mpls traffic-eng tunnels tunnel1**, **show isis fast-reroute ti-lfa tunnel**, **show frr-manager client client-name ISIS interfaces detail**, **show ip cef 6.6.6.6 internal** |

## Restrictions for Segment Routing-Traffic Engineering With IS-IS

- SR-TE is not supported on broadcast interfaces; it is supported only point-to-point interfaces.
- Only one instance of protocol should be enabled for TE at a given point of time.

- You can use the verbatim keyword only on a label-switched path (LSP) that is configured with the explicit path option.
- Re-optimization is unsupported on the verbatim LSP.

# Information About Segment Routing Traffic Engineering With IS-IS

A Traffic Engineered (TE) tunnel is a container of TE LSPs instantiated between the tunnel ingress and the tunnel destination. A TE tunnel can instantiate one or more SR-TE LSPs that are associated with the same tunnel. The SR-TE LSP path may not necessarily follow the same IGP path to a destination node. In this case, the SR-TE path can be specified through either a set of prefix-SIDs, or adjacency-SIDs of nodes, or both, and links to be traversed by the SR-TE LSP.

The head-end imposes the corresponding MPLS label stack on outgoing packets to be carried over the tunnel. Each transit node along the SR-TE LSP path uses the incoming top label to select the next-hop, pop or swap the label, and forward the packet to the next node with the remainder of the label stack, until the packet reaches the ultimate destination. The set of hops or segments that define an SR-TE LSP path are provisioned by the operator.

## SR-TE LSP Instantiation

A Traffic Engineered (TE) tunnel is a container of one or more instantiated TE LSPs. An SR-TE LSP is instantiated by configuring ‘segment-routing’ on the path-option of the TE tunnel. The traffic mapped to the tunnel is forwarded over the primary SR-TE instantiated LSP.

Multiple path-options can also be configured under the same tunnel. Each path-option is assigned a preference index or a path-option index that is used to determine the more favorable path-option for instantiating the primary LSP—the lower the path-option preference index, the more favorable the path-option. The other less favorable path-options under the same TE tunnel are considered secondary path-options and may be used once the currently used path-option is invalidated (for example, due to a failure on the path.

**Note** A forwarding state is maintained for the primary LSP only.

### SR-TE LSP Explicit Null

MPLS-TE tunnel head-end does not impose explicit-null at the bottom of the stack. When penultimate hop popping (PHP) is enabled for SR prefix SIDs or when an adjacency SID is the last hop of the SR-TE LSP, the packet may arrive at the tail-end without a transport label. However, in some cases, it is desirable that the packet arrive at the tail-end with explicit-null label, and in such case, the head-end will impose an explicit-null label at the top of the label stack.

### SR-TE LSP Path Verification

SR-TE tunnel functionality requires that the head-end perform initial verification of the tunnel path as well as the subsequent tracking of the reachability of the tunnel tail-end and traversed segments.

Path verification for SR-TE LSP paths is triggered whenever MPLS-TE is notified of any topology changes or SR SID updates.

The SR-TE LSP validation steps consist of the following checks:

### Topology Path Validation

The head-end validates the path of an SR-TE LSP for connectivity against the TE topology. MPLS-TE head-end checks if links corresponding to the adjacency SIDs are connected in the TE topology.

For newly-instantiated SR-TE LSPs, if the head-end detects a discontinuity on any link of the SR-TE path, that path is considered invalid and is not used. If the tunnel has other path-options with valid paths, those paths are used to instantiate the tunnel LSP.

For TE tunnels with existing instantiated SR-TE LSP, if the head-end detects a discontinuity on any link, the head-end assumes a fault has occurred on that link. In this case, the local repair protection, such as the IP FRR, come in to effect. The IGPs continue to sustain the protected adjacency label and associated forwarding after the adjacency is lost for some time. This allows the head-ends enough time to reroute the tunnels onto different paths that are not affected by the same failure. The head-end starts a tunnel invalidation timer once it detects the link failure to attempt to reroute the tunnel onto other available path-options with valid paths.

If the TE tunnel is configured with other path-options that are not affected by the failure and are validated, the head-end uses one of those path-options to reroute (and re-optimize) the tunnel by instantiating a new primary LSP for the tunnel using the unaffected path.

If no other valid path-options exist under the same tunnel, or if the TE tunnel is configured with only one path-option that is affected by the failure, the head-end starts an invalidation timer after which it brings the tunnel state to ‘down’. This action avoids a null route from being sent along with traffic flowing over the affected SR-TE LSP, and allows services riding over the tunnel to reroute over different available paths at the head-end. There is an invalidation drop configuration that keeps the tunnel 'up', but drops the traffic when the invalidation timer expires.

For intra-area SR-TE LSPs, the head-end has full visibility over the LSP path, and validates the path to the ultimate LSP destination. However, for inter-area LSPs, the head-end has partial visibility over the LSP path—only up to the first ABR. In this case, the head-end can only validate the path from the ingress to the first ABR. Any failure along the LSP beyond the first ABR node is invisible to the head-end, and other mechanisms to detect such failures, such as BFD over LSP are assumed.

### SR SID Validation

SID hops of an SR-TE LSP are used to determine the outgoing MPLS label stack to be imposed on the outgoing packets carried over the SR-TE LSP of a TE tunnel. A database of global and local adjacency-SIDs is populated from the information received from IGPs and maintained in MPLS-TE. Using a SID that is not available in the MPLS TE database invalidates the path-option using the explicit-path. The path-option, in this case, is not used to instantiate the SR TE LSP. Also, withdrawing, adding, or modifying a SID in the MPLS-TE SID-database, results in the MPLS-TE head-end verifying all tunnels with SR path-options (in-use or secondary) and invokes proper handling.

### LSP Egress Interface

When the SR-TE LSP uses an adjacency-SID for the first path hop, TE monitors the interface state and IGP adjacency state associated with the adjacency-SID and the node that the SR-TE LSP egresses on. If the interface or adjacency goes down, TE can assume a fault occurred on the SR-TE LSP path and take the same reactive actions described in the previous sections.

**Note** When the SR-TE LSP uses a prefix-SID for the first hop, TE cannot directly infer on which interface the tunnel egresses. TE relies on the IP reachability information of the prefix to determine if connectivity to the first hop is maintained.

### IP Reachability Validation

MPLS-TE validates that the nodes corresponding to the prefix-SIDs are IP reachable before declaring the SR path valid. MPLS-TE detects path changes for the IP prefixes corresponding to the adjacency or prefix SIDs of the SR-TE LSP path. If the node announcing a specific SID loses IP reachability. due to a link or node failure, MPLS-TE is notified of the path change (no path). MPLS-TE reacts by invalidating the current SR-TE LSP path, and may use other path-options with a valid path, if any to instantiate a new SR-TE LSP.

**Note** Since IP-FRR does not offer protection against failure of a node that is being traversed by an SR-TE LSP (such as, a prefix-SID failure along the SR-TE LSP path), the head-end immediately reacts to IP route reachability loss for prefix-SID node by setting the tunnel state to ‘down’ and removes the tunnel forwarding entry if there are no other path-options with valid path for the affected tunnel.

### Tunnel Path Affinity Validation

The affinity of a tunnel path can be specified using the command **tunnel mpls traffic-eng affinity** under the tunnel interface.

The head-end validates that the specified SR path is compliant with the configured affinity. This necessitates that the paths of each segment of the SR path be validated against the specified constraint. The path is declared invalid against the configured affinity constraints if at least a single segment of the path does not satisfy the configured affinity.

### Tunnel Path Resource Avoidance Validation

You can specify a set of addresses to be validated as excluded from being traversed by SR-TE tunnel packets. To achieve this, the head-end runs the per-segment verification checks and validates that the specified node, prefix or link addresses are indeed excluded from the tunnel in the SR path. The tunnel resource avoidance checks can be enabled per path using the commands below. The list of addresses to be excluded are defined and the name of the list is referenced in the path-option.

```
interface tunnel100
 tunnel mpls traffic-eng path-option 1 explicit name EXCLUDE segment-routing
ip explicit-path name EXCLUDE enable
 exclude-address 192.168.0.2
 exclude-address 192.168.0.4
 exclude-address 192.168.0.3
!
```

### Verbatim Path Support

MPLS TE LSPs usually require that all the nodes in the network are TE aware which means that they have IGP extensions to TE in place. However, some network administrators want the ability to build TE LSPs to traverse nodes that do not support IGP extensions to TE, but that do support RSVP extensions to TE. Verbatim LSPs are helpful when all or some of the intermediate nodes in a network do not support IGP extensions for TE.

When this feature is enabled, the IP explicit path is not checked against the TE topology database. Since the TE topology database is not verified, a Path message with IP explicit path information is routed using the shortest path first (SPF) algorithm for IP routing.

## SR-TE Traffic Load Balancing

SR-TE tunnels support the following load-balancing options:

### Load Balancing on Port Channel TE Links

Port Channel interfaces carry the SR-TE LSP traffic. This traffic load balances over port channel member links as well as over bundle interfaces on the head or mid of an SR-TE LSP.

### Load Balancing on Single Tunnel

While using the equal cost multi path protocol (ECMP), the path to a specific prefix-SID may point to multiple next-hops. And if the SR-TE LSP path traverses one or more prefix-SIDs that have ECMP, the SR-TE LSP traffic load-balances on the ECMP paths of each traversed prefix-SID from the head-end or any midpoint traversed node along the SR-TE LSP path.

### Load Balancing on Multiple Tunnels

Multiple TE tunnels can be used as next-hop paths for routes to specific IP prefixes either by configuring static route on multiple tunnels, or auto-route announcing multiple parallel tunnels to the same destination. In such cases, the tunnels share the traffic load equally or load balance traffic on multiple parallel tunnels. It is also possible to allow Unequal Load Balance (UELB) with an explicit per tunnel configuration at the tunnel head-end. In this case, the tunnel load-share is passed from MPLS-TE to forwarding plane.

The tunnel load-share feature continues to work for TE tunnels that instantiate the SR-TE LSPs.

## SR-TE Tunnel Re-optimization

TE tunnel re-optimization occurs when the head-end determines that there is a more optimal path available than the one currently used. For example, in case of a failure along the SR-TE LSP path, the head-end could detect and revert to a more optimal path by triggering re-optimization.

Tunnels that instantiate SR-TE LSP can re-optimize without affecting the traffic carried over the tunnel.

Re-optimization can occur because:

- the explicit path hops used by the primary SR-TE LSP explicit path are modified,
- the head-end determines the currently used path-option are invalid due to either a topology path disconnect, or a missing SID in the SID database that is specified in the explicit-path
- a more favorable path-option (lower index) becomes available

When the head-end detects a failure on a protected SR adjacency-SID that is traversed by an SR-TE LSP, it starts the invalidation timer. If the timer expires and the head-end is still using the failed path because it is unable to reroute on a different path, the tunnel state is brought ‘down’ to avoid a null route from being sent along with the traffic. Once the tunnel is down, services on the tunnel converge to take a different path.

The following is a sample output of a manual re-optimization example. In this example, the path-option is changed from ‘10’ to ‘20’.

```
Router# mpls traffic-eng reoptimize tunnel 1 path-option 20
The targeted path-option is not in lock down mode. Continue? [no]: yes
Router# show mpls traffic-eng tunnels tunnel1
```

```
Name: R1_t1                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 20, (SEGMENT-ROUTING) type explicit IP_PATH (Basis for Setup)
    path option 10, (SEGMENT-ROUTING) type dynamic
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 20 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours, 9 minutes
      Time since path change: 14 seconds
      Number of LSP IDs (Tun_Instances) used: 1819
    Current LSP: [ID: 1819]
      Uptime: 17 seconds
      Selection: reoptimization
    Prior LSP: [ID: 1818]
      ID: path option unknown
      Removal Trigger: reoptimization completed
  Tun_Instance: 1819
  Segment-Routing Path Info (isis level-1)
    Segment0[Node]: 4.4.4.4, Label: 114
    Segment1[Node]: 5.5.5.5, Label: 115
    Segment2[Node]: 6.6.6.6, Label: 116
```

## SR-TE With Lockdown Option

The **lockdown** option prevents SR-TE from re-optimizing to a better path. However, it does not prevent signaling the existence of a new path.

```
interface Tunnel1
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng path-option 10 segment-routing lockdown
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10                      (Tunnel1) Destination: 6.6.6.6

  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (LOCKDOWN) type segment-routing (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: enabled  Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: segment-routing path option 10 is active
```

```
  BandwidthOverride: disabled  LockDown: enabled   Verbatim: disabled
History:
  Tunnel:
    Time since created: 6 days, 19 hours, 22 minutes
    Time since path change: 1 minutes, 26 seconds
    Number of LSP IDs (Tun_Instances) used: 1822
  Current LSP: [ID: 1822]
    Uptime: 1 minutes, 26 seconds
    Selection: reoptimization
  Prior LSP: [ID: 1821]
    ID: path option unknown
    Removal Trigger: configuration changed
Tun_Instance: 1822
Segment-Routing Path Info (isis  level-1)
  Segment0[Node]: 6.6.6.6, Label: 116
```

## SR-TE Tunnel Protection

Protection for SR TE tunnels can take any of the following alternatives:

### IP-FRR Local Repair Protection

On an SR-TE LSP head-end or mid-point node, IP-FRR is used to compute and program the backup protection path for the prefix-SID or adjacency-SID label.

With IP-FRR, backup repair paths are pre-computed and pre-programmed by IGPs *before* a link or node failure. The failure of a link triggers its immediate withdrawal from the TE topology (link advertisement withdrawal). This allows the head-end to detect the failure of an SR-TE LSP traversing the failed adjacency-SID.

When a protected adjacency-SID fails, the failed adjacency-SID label and associated forwarding are kept functional for a specified period of time (5 to 15 minutes) to allow all SR TE tunnel head-ends to detect and react to the failure. Traffic using the adjacency-SID label continues to be FRR protected even if there are subsequent topology updates that change the backup repair path. In this case, the IGPs update the backup repair path while FRR is active to reroute traffic on the newly-computed backup path.

When the primary path of a protected prefix-SID fails, the PLR reroutes to the backup path. The head-end remains transparent to the failure and continues to use the SR-TE LSP as a valid path.

IP-FRR provides protection for adjacency and prefix-SIDs against link failures only.

### Tunnel Path Protection

Path protection is the instantiation of one or more standby LSPs to protect against the failure of the primary LSP of a single TE tunnel.

Path protection protects against failures by pre-computing and pre-provisioning secondary paths that are failure diverse with the primary path-option under the same tunnel. This protection is achieved by computing a path that excludes prefix-SIDs and adjacency-SIDs traversed by the primary LSP or by computing a path that excludes SRLGs of the primary SR-TE LSP path.

In the event of a failure of the primary SR-TE LSP, at least one standby SR-TE LSP is used for the tunnel. Multiple secondary path-options can be configured to be used as standby SR-TE LSPs paths.

# Unnumbered Support

IS-IS description of an unnumbered link does not contain remote interface ID information. The remote interface ID of an unnumbered link is required to include the unnumbered link as part of the SR-TE tunnel.

# How to Configure Segment Routing Traffic Engineering With IS-IS

Perform the following steps to configure Segment Routing Traffic Engineering (SR-TE) with IS-IS.

## Configuring Path Option for a TE Tunnel

The **segment-routing** keyword indicates that the specified path is programmed as an SR path:

```
Device(config)# interface tunnel 100
Device(config-if)# tunnel mpls traffic-eng path-option 1 explicit name foo segment-routing
Device(config-if)# tunnel mpls traffic-eng path-option 2 dynamic segment-routing
Device(config-if)# tunnel mpls traffic-eng path-option 3 segment-routing
```

**Note** With IP unnumbered interfaces dynamic path is not supported.

When the path-option type for an operational SR tunnel is changed from SR to non-SR (for example, **dynamic**), the existing forwarding entry of the tunnel is deleted.

Segment Routing can be enabled or disabled on an existing secondary or an in-use path-option. If the tunnel uses a signaled RSVP-TE explicit path-option and segment routing is enabled on that tunnel, the RSVP-TE LSP is torn, and the SR-TE LSP is instantiated using the same path-option. Conversely, if segment routing is disabled on a path-option that is in use by the primary LSP, the tunnel goes down intermittently and a new RSVP-TE LSP will be signaled using the same explicit path.

If the segment-routing path-option is enabled on a secondary path-option (that is, not in-use by the tunnel's primary LSP), the tunnel is checked to evaluate if the newly specified SR-TE LSP path-option is valid and more favorable to use for the tunnel primary LSP.

## Configuring SR Explicit Path Hops

The following SR-TE explicit path hops are supported:

- IP addresses
- MPLS labels
- Mix of IP addresses and MPLS labels

For intra-area LSPs, the explicit path can be specified as a list of IP addresses.

```
Device(config)# ip explicit-path name foo
Device(config-ip-expl-path)# index 10 next-address 1.1.1.1 node address
Device(config-ip-expl-path)# index 20 next-address 12.12.12.2 link address
```

**Note** When using IP unnumbered interfaces, you cannot specify next hop address as an explicit path index. It should be node address or label.

The explicit path can also be specified as segment-routing SIDs:

```
Device(config)# ip explicit-path name foo
Device(config-ip-expl-path)# index 10 next-label 20
```

**Note** IP addresses cannot be used after using the label in MIXED_PATH.

# Configuring Affinity on an Interface

Perform the following steps to configure affinity on an interface:

```
interface GigabitEthernet2
 ip address 192.168.2.1 255.255.255.0
 ip router isis 1
 negotiation auto
 mpls traffic-eng tunnels
 mpls traffic-eng attribute-flags 0x1
 isis network point-to-point
 ip rsvp bandwidth
```

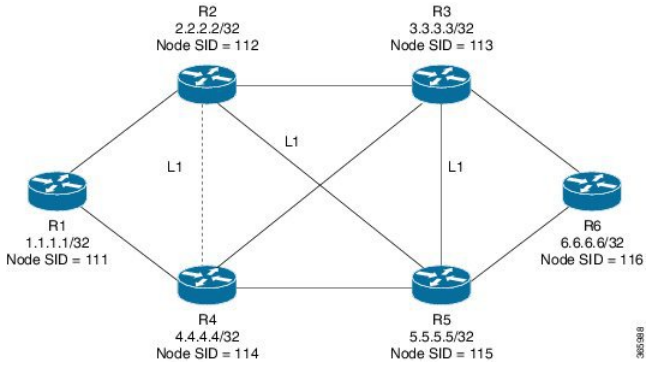
# Enabling Verbatim Path Support

To enable verbatim with SR-TE you can use the following example. In the example, tunnel destination 11.11.11.11 is in different area and an explicit path with name multihop is defined with SR-TE path option.

```
R6#
interface Tunnel4
ip unnumbered Loopback66
tunnel mode mpls traffic-eng
tunnel destination 11.11.11.11
tunnel mpls traffic-eng path-option 1 explicit name multihop segment-routing verbatim
!
ip explicit-path name multihop enable
index 1 next-label 16003
index 2 next-label 16002
index 3 next-label 16001
!
End
```

# Use Case: Segment Routing Traffic Engineering Basic Configuration

Consider the following topology to understand the SR-TE configuration:




To configure at the head-end router, R1:

```
!
mpls traffic-eng tunnels
!
segment-routing mpls
connected-prefix-sid-map
  address-family ipv4
   1.1.1.1/32 index 111  range 1
  exit-address-family
!
set-attributes
  address-family ipv4
  sr-label-preferred
exit-address-family
!
interface Loopback1
ip address 1.1.1.1 255.255.255.255
ip router isis 1
!
int gig0/0
ip address 11.11.11.1 255.255.255.0
ip router isis 1
mpls traffic-eng tunnels
isis network point-to-point
!
router isis 1
net 49.0001.0010.0100.1001.00
is-type level-1
metric-style wide
segment-routing mpls
segment-routing prefix-sid-map advertise-local
mpls traffic-eng router-id  Loopback1
mpls traffic-eng level-1
!
end
```

To enable SR-TE Explicit path (Node SID based), enable the following CLI on R1:

```
Head end SR-TE configuration R1#
!
interface tunnel1
ip unnumbered Loopback1
tunnel mode mpls traffic-eng
tunnel destination 6.6.6.6
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng priority 6 6
```

```
tunnel mpls traffic-eng path-option 10 explicit name Node_PATH segment-routing
!
ip explicit-path name Node_PATH
 next-label  16114
next-label  16115
next-label  16116
```

To verify proper operation of SR-TE tunnel 1 on R1 enable the following CLI:

```
Tunnel verification on (R1)# show mpls traffic-eng tun tun 1 detail
Name: R1_t1                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type explicit Node_PATH (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
Verbatim: disabled
Number of LSP IDs (Tun_Instances) used: 1815
    Current LSP: [ID: 1815]
      Uptime: 2 seconds
Removal Trigger: configuration changed
    Segment-Routing Path Info (isis  level-1)
    Segment0[Node]: 4.4.4.4, Label: 16114
    Segment1[Node]: 5.5.5.5, Label: 16115
    Segment2[Node]: 6.6.6.6, Label: 16116
```

To configure at the tail-end router, R6:

```
interface GigabitEthernet2
ip address 100.101.1.1 255.255.255.0
ip router isis 1
negotiation auto
mpls traffic-eng tunnels
router isis 1
 net 49.0001.0060.0600.6006.00
 ispf level-1
 metric-style wide
 log-adjacency-changes
 segment-routing mpls

segment-routing prefix-sid-map advertise-local
 mpls traffic-eng router-id Loopback1
 mpls traffic-eng level-1
```

## Explicit Path SR-TE Tunnel 1

Consider tunnel 1 based only on IP addresses:

```
ip explicit-path name IP_PATH1
 next-address 2.2.2.2
 next-address 3.3.3.3
 next-address 6.6.6.6
!
interface Tunnel1
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng path-option 10 explicit name IP_PATH1 segment-routing
 tunnel mpls traffic-eng path-selection metric igp
```

```
 tunnel mpls traffic-eng load-share 10
end
```

## Explicit Path SR-TE Tunnel 2

Consider tunnel 2 based on node SIDs

```
ip explicit-path name IA_PATH
 next-label 114
 next-label 115
 next-label 116
!
interface Tunnel2
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng bandwidth 10000 class-type 1
 tunnel mpls traffic-eng path-option 10 explicit name NODE_PATH segment-routing
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10
end
```

## Explicit Path SR-TE Tunnel 3

Consider that tunnel 3 is based on a mix of IP addresses and label

```
ip explicit-path name MIXED_PATH enable
 next-address 2.2.2.2
 next-address 3.3.3.3
 next-label 115
 next-label 116
!
interface Tunnel3
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng path-option 10 explicit name MIXED_PATH segment-routing
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10
```

**Note** In the case of mixed path, IP next-hop cannot be used after using Node SIDs in the path. The following path will not be valid:

```
ip explicit-path name MIXED_PATH enable
next-label 115
next-label 116
next-address 2.2.2.2
```

## Dynamic Path SR-TE Tunnel 4

Consider that tunnel 4is based on adjacency SIDs

```
interface Tunnel4
```

```
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng bandwidth 10000 class-type 1
 tunnel mpls traffic-eng path-option 10 dynamic segment-routing
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10
end
```

## Dynamic Path SR-TE Tunnel 5

Consider that tunnel 5 is based on Node SIDs

```
 interface Tunnel5
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng path-option 10 segment-routing
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10
```

# Verifying Configuration of the SR-TE Tunnels

Use the **show mpls traffic-eng tunnels** *tunnel-number* command to verify the configuration of the SR-TE tunnels.

## Verifying Tunnel 1

```
Name: R1_t1                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type explicit IP_PATH (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours
      Time since path change: 2 seconds
      Number of LSP IDs (Tun_Instances) used: 1814
    Current LSP: [ID: 1814]
      Uptime: 2 seconds
      Selection: reoptimization
    Prior LSP: [ID: 1813]
      ID: path option unknown
      Removal Trigger: configuration changed
```

```
  Tun_Instance: 1814
  Segment-Routing Path Info (isis  level-1)
    Segment0[Node]: 4.4.4.4, Label: 114
    Segment1[Node]: 5.5.5.5, Label: 115
    Segment2[Node]: 6.6.6.6, Label: 116
```

## Verifying Tunnel 2

```
Name: R1_t2                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type explicit IA_PATH (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours, 1 minutes
      Time since path change: 1 seconds
      Number of LSP IDs (Tun_Instances) used: 1815
    Current LSP: [ID: 1815]
      Uptime: 1 seconds
    Prior LSP: [ID: 1814]
      ID: path option unknown
      Removal Trigger: configuration changed
  Tun_Instance: 1815
  Segment-Routing Path Info (isis  level-1)
    Segment0[ - ]: Label: 114
    Segment1[ - ]: Label: 115
    Segment2[ - ]: Label: 116
```

## Verifying Tunnel 3

```
Name: R1_t3                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type explicit MIXED_PATH (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours, 2 minutes
```

```
    Time since path change: 2 seconds
    Number of LSP IDs (Tun_Instances) used: 1816
  Current LSP: [ID: 1816]
    Uptime: 2 seconds
    Selection: reoptimization
  Prior LSP: [ID: 1815]
    ID: path option unknown
    Removal Trigger: configuration changed
Tun_Instance: 1816
Segment-Routing Path Info (isis  level-1)
  Segment0[Node]: 2.2.2.2, Label: 112
  Segment1[Node]: 3.3.3.3, Label: 113
  Segment2[ - ]: Label: 115
  Segment3[ - ]: Label: 116
```

## Verifying Tunnel 4

```
Name: R1_t4                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type dynamic (Basis for Setup, path weight 30)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: dynamic path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours
      Time since path change: 2 seconds
      Number of LSP IDs (Tun_Instances) used: 1813
    Current LSP: [ID: 1813]
      Uptime: 2 seconds
    Prior LSP: [ID: 1806]
      ID: path option unknown
      Removal Trigger: configuration changed
  Tun_Instance: 1813
  Segment-Routing Path Info (isis  level-1)
    Segment0[Link]: 192.168.2.1 - 192.168.2.2, Label: 17
    Segment1[Link]: 192.168.4.2 - 192.168.4.1, Label: 25
    Segment2[Link]: 192.168.8.1 - 192.168.8.2, Label: 300
```

## Verifying Tunnel 5

```
Name: R1_t5                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, type segment-routing (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
```

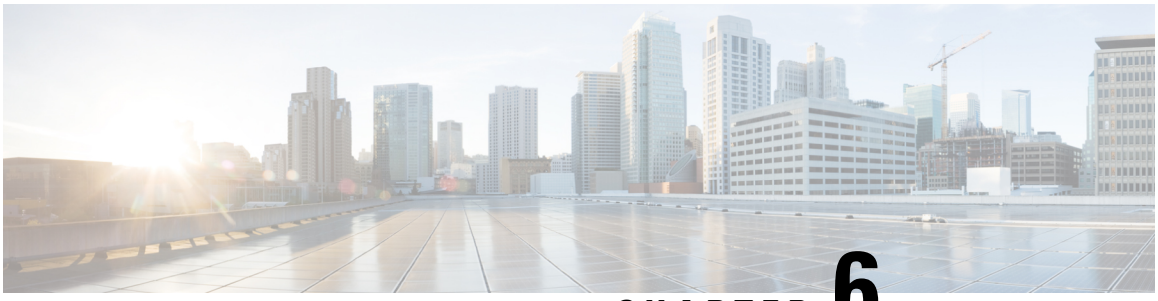
```
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: segment-routing path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours, 4 minutes
      Time since path change: 14 seconds
      Number of LSP IDs (Tun_Instances) used: 1817
    Current LSP: [ID: 1817]
      Uptime: 14 seconds
      Selection: reoptimization
    Prior LSP: [ID: 1816]
      ID: path option unknown
      Removal Trigger: configuration changed
  Tun_Instance: 1817
  Segment-Routing Path Info (isis  level-1)
    Segment0[Node]: 6.6.6.6, Label: 116
```

# Verifying Verbatim Path Support

To verify proper operation and SR-TE tunnel state use following CLI:

```
R6#sh mpl traffic-eng tunnels tunnel 4

Name: R6_t4                                (Tunnel4) Destination: 11.11.11.11
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 1, (SEGMENT-ROUTING) type explicit (verbatim) multihop (Basis for Setup)

  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    Path Selection:
     Protection: any (default)
    Path-selection Tiebreaker:
      Global: not set   Tunnel Specific: not set   Effective: min-fill (default)
    Hop Limit: disabled [ignore: Verbatim Path Option]
    Cost Limit: disabled
    Path-invalidation timeout: 10000 msec (default), Action: Tear
    AutoRoute: disabled LockDown: disabled Loadshare: 0 [0] bw-based
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 1 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: enabled

  History:
    Tunnel:
      Time since created: 16 minutes, 40 seconds
      Time since path change: 13 minutes, 6 seconds
      Number of LSP IDs (Tun_Instances) used: 13
    Current LSP: [ID: 13]
      Uptime: 13 minutes, 6 seconds
      Selection: reoptimization
    Prior LSP: [ID: 12]
      ID: path option unknown
      Removal Trigger: configuration changed (severe)
  Tun_Instance: 13
  Segment-Routing Path Info (IGP information is not used)
```

```
Segment0[First Hop]: 0.0.0.0, Label: 16003
Segment1[ - ]: Label: 16002
Segment2[ - ]: Label: 16001
```



CHAPTER 6

# Segment Routing With OSPFv2 Node SID

This chapter describes how Segment Routing works with OSPFv2 node SID.


- Feature Information for Segment Routing With OSPFv2 Node SID, on page 59
- Information About Segment Routing With OSPFv2 Node SID, on page 59
- How to Configure Segment Routing With OSPFv2 Node SID, on page 62
- Additional References for Segment Routing With OSPFv2 Node SID, on page 70


## Feature Information for Segment Routing With OSPFv2 Node SID

*Table 5: Feature Information for Segment Routing With OSPFv2 Node SID*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing With OSPF | Cisco IOS XE Amsterdam 17.3.2 | The Segment Routing OSPFv2 node SID feature provides support for segment routing on OSPF networks.<br>The following commands were introduced or modified: **connected-prefix-sid-map**, **show ip ospf 10 segment-routing**, **sr-label-preferred**, **ip ospf prefix-attributes n-flag-clear** . |

## Information About Segment Routing With OSPFv2 Node SID

Segment Routing relies on a small number of extensions to Open Shortest Path First (OSPF) protocols. There are two levels of configuration required to enable segment routing for a routing protocol instance. The top level segment routing configuration which is managed by segment routing infrastructure component enables segment routing, whereas, segment routing configuration at the router ospf level enables segment routing for the ospf instance. There are three segment routing states:

- SR_NOT_CONFIGURED
- SR_DISABLED
- SR_ENABLED

Segment routing configuration under the IGPs is allowed only if the SR state is either SR_DISABLED or SR_ENABLED. The SR_ENABLED state indicates that there is at least a valid SRGB range reserved. You can enable segment routing for IGPs under the router configuration sub mode, through commands. However, IGP segment routing are enabled only after the global SR is configured.

The SR_ENABLED is a necessary state for any protocol to enable SR, however, it is not a sufficient for enabling SR for a protocol instance. The reason being that the OSPF still does not have any information about segment routing global block (SRGB) information. When the request to receive information about the SRGB is processed successfully, the OSPF SR operational state is enabled.

Segment Routing requires each router to advertise its segment routing data-plane capability and the range of MPLS label values that are used for segment routing in the case where global SIDs are allocated. Data-plane capabilities and label ranges are advertised using the SR-capabilities sub-TLV inserted into the OSPF Router Information Opaque LSA.

OSPF SR-capabilities sub TLV includes all reserved SRGB ranges. However, the Cisco implementation supports only one SRGB range.

# Prefix-SID Received in Label Switched Path From Remote Routers

OSPF sends the prefix SIDs associated with the connected prefix using the Extended Prefix Sub TLV in its opaque Extended prefix LSA. Prefix SIDs received in a LSA which have got reachability are downloaded to the routing information base (RIB) in the same way as BGP downloads per prefix VPN labels, only if the following conditions are met:

- Segment routing is enabled for the topology and address-family.
- Prefix-SID is valid.
- The local label binding to MFI is successful.

**Note** For SIDs that do not fit in the specified SID range, labels are not used when updating the RIB. For the cases, where SID fits in the SID range, but does not fit the next-hop neighbor SID range, remote label associated with that path is not installed.

# Segment Routing Adjacency SID Advertisement

Effective with Cisco IOS-XE Release 3.17, OSPF supports the advertisement of segment routing adjacency SID. An Adjacency Segment Identifier (Adj-SID) represents a router adjacency in Segment Routing.

A segment routing-capable router may allocate an Adj-SID for each of its adjacencies and an Adj-SID sub-TLV is defined to carry this SID in the Extended Opaque Link LSA.

OSPF allocates the adjacency SID for each OSPF neighbor if the OSPF adjacency which are in two way or in FULL state. OSPF allocates the adjacency SID only if the Segment Routing is enabled. The label for adjacency SID is dynamically allocated by the system. This eliminates the chances of misconfiguration, as this has got only the local significance.

### Multiple Adjacency-SIDs

Effective with Cisco IOS-XE Release 16.3, multiple adjacency-SIDs are supported. For each OSPF adjacency, OSPF allots to Adj SIDs, unprotected and protected Adj-SIDs which are carried in the extended link LSAs. The protected adjacency SID (or back up Adj-SID) is allocated and advertised only when FRR is enabled on the router and also on the interface where SR is enabled on the system. When FRR or SR is disabled, the protected Adj-SID is released.

The persistence of protected adj-SID in forwarding plane is supported. When the primary link is down, OSPF delays the release of its backup Adj-SID until the delay timer (30 sec) expires. This allows the forwarding plane to continue to forward the traffic through the backup path until the router is converged.

The allocated and advertised backup Adj-SIDs can be displayed in the output of **show ip ospf neighbor detail** and **show ip ospf segment-routing protected-adjacencies command**.

## Segment Routing Mapping Server

Segment Routing Mapping Server (SRMS) allows configuration and maintenance of the Prefix-SID mapping policy entries. Effective with Cisco IOS-XE Release 3.17, the IGPs use the active policy of the SRMS to determine the SID values when programming the forwarding plane.

The SRMS provides prefixes to SID/Label mapping policy for the network. IGPs, on the other hand, are responsible for advertising prefixes to SID/Label mapping policy through the Prefix-SID/Label Binding TLV.

Active policy information and changes are notified to the IGPs, which use active policy information to update forwarding information.

### Connected Prefix SIDs

When a router installs a prefix with a SID that is different than what it advertises to the LSP, for example, if more than one protocol or more than one IGP instance is announcing the same prefix with different SIDs to the SRMS, the SRMS resolves the conflict and announces the winning prefix and SID that may not be the same as the local instance. In that case, the IGP always advertises what it learns from its source LSP although it still tries to install the SID which may be different than what it learns in its LSP. This is done to prevent the IGP from redistributing the SIDs from another protocol or another protocol instance.

## SRGB Range Changes

When OSPF segment routing is configured, OSPF must request an interaction with the SRGB before OSPF SR operational state can be enabled. If no SRGB range is created, OSPF will not be enabled.

When an SRGB change event occurs, OSPF makes the corresponding changes in its sub-block entries. OSPF also advertises the newly created or extended SRGB range in SR-capabilities sub-TLV and updates the prefix-sid sub TLV advertisement.

## MPLS Forwarding on an Interface

MPLS forwarding must be enabled before segment routing can use an interface. OSPF is responsible for enabling MPLS forwarding on an interface.

When segment routing is enabled for a OSPF topology, or OSPF segment routing operational state is enabled, it enables MPLS for any interface on which the OSPF topology is active. Similarly, when segment routing is disabled for a OSPF topology, it disables the MPLS forwarding on all interfaces for that topology.

## Conflict Handling of SID Entries

When there is a conflict between the SID entries and the associated prefix entries use any of the following methods to resolve the conflict:

- When the system receives two SID entries for the same prefix, then the prefix received by higher router ID is treated as the SID corresponding to the prefix. The prefix is installed with the SID entry which was advertised by the higher router ID.
- When the system receives two SID entries one by OSPF protocol and the other by IS-IS protocol, then the SID entry received by OSPF protocol is treated as valid SID. The prefix is installed with the SID entry which was received by OSPF protocol.
- When two prefixes are advertised with the same SID entry, then the prefix which is advertised by the higher router ID is installed with the SID entry and the other prefix is installed without any SID entry.

In an ideal situation, each prefix should have unique SID entries assigned.

# How to Configure Segment Routing With OSPFv2 Node SID

Perform the following steps to configure segment routing with OSPFv2 node SID.

## Configuring Segment Routing With OSPF

### Before you begin

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

### SUMMARY STEPS

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

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device# enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | `Device# configure terminal` | |
| **Step 3** | segment-routing mpls<br>**Example:**<br>`Device(config-sr)# segment-routing mpls` | Enables the segment feature using the mpls data plane. |
| **Step 4** | connected-prefix-sid-map<br>**Example:**<br>`Device(config-srmpls)# connected-prefix-sid-map` | Enters a sub-mode where you can configure address-family specific mappings for local prefixes and SIDs. |
| **Step 5** | address-family ipv4<br>**Example:**<br>`Device(config-srmpls-conn)# address-family ipv4` | Specifies IPv4 address prefixes. |
| **Step 6** | 1.1.1.1/32 index 100 range 1<br>**Example:**<br>`Device(config-srmpls-conn-af)# 1.1.1.1/32 100 range 1` | Associates SID 100 with the address 1.1.1.1/32. |
| **Step 7** | exit-address-family<br>**Example:**<br>`Device(config-srmpls-conn-af)# exit-address-family` | Exits the address family. |

# Configuring Segment Routing on OSPF Network

### Before you begin

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

### SUMMARY STEPS

1. **router ospf 10**
2. **router-id**<*id*>
3. **segment-routing mpls**
4. **segment-routing area** <*area id*> **mpls**
5. **show ip ospf 10 segment-routing**

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **router ospf 10**<br>**Example:**<br>`Device(config)# router ospf 10` | Enables the OSPF mode. |
| **Step 2** | **router-id**<*id*><br>**Example:**<br>`Device(config-router)# router-id 1.0.0.0` | Configures OSPF routes. |
| **Step 3** | **segment-routing mpls**<br>**Example:**<br>`Device(config-router)# segment-routing mpls` | Configures segment routing mpls mode. |
| **Step 4** | **segment-routing area** <*area id*> **mpls**<br>**Example:**<br>`Device(config-router) # segment-routing area 0 mpls` | Configures segment routing mpls mode in a specific area. |
| **Step 5** | **show ip ospf 10 segment-routing**<br>**Example:**<br>`Device# show ip ospf 10 segment-routing` | Shows the output for configuring SR under OSPF.<br>The following example displays output from the show ip ospf segment-routing state command for the segment routing under OSPF:<br>`Device#show ip ospf 10 segment-routing`<br><br>`OSPF Router with ID (0.0.0.1) (Process ID 10)`<br><br>`Global segment-routing state: Enabled`<br><br>`Segment Routing enabled:`<br>`Area Topology name Forwarding`<br>`0 Base MPLS`<br>`1 Base MPLS`<br><br>`SR Attributes`<br>`Prefer non-SR (LDP) Labels`<br>`Do not advertise Explicit Null`<br><br>`Local MPLS label block (SRGB):`<br>`Range: 16000 - 23999`<br>`State: Created`<br><br>`Registered with SR App, client handle: 3`<br>`Connected map notifications active (handle 0x4), bitmask 0x1`<br>`Active policy map notifications active (handle 0x5), bitmask 0xC`<br>`Registered with MPLS, client-id: 100` |

| | Command or Action | Purpose |
|---|---|---|
| | | `Bind Retry timer not running`<br>`Adj Label Bind Retry timer not running` |

# Configuring Prefix-SID for OSPF

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

### Before you begin

Segment routing must be enabled on the corresponding address family.

### SUMMARY STEPS

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

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | enable<br>**Example:**<br>`Device# enable` | Enables privileged EXEC mode. |
| **Step 2** | configure terminal<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | segment-routing mpls<br>**Example:**<br>`Device(config)# segment-routing mpls` | Configures segment routing mpls mode. |
| **Step 4** | connected-prefix-sid-map<br>**Example:**<br>`Device(config-srmpls)# connected-prefix-sid-map` | Enters a sub-mode where you can configure address-family specific mappings for local prefixes and SIDs. |
| **Step 5** | address-family ipv4<br>**Example:** | Specifies the IPv4 address family and enters router address family configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | `Device(config-srmpls-conn)# address-family ipv4` | |
| **Step 6** | 1.1.1.1/32 index 100 range 1<br>**Example:**<br>`Device(config-srmpls-conn-af)# 1.1.1.1/32 100 range 1` | Associates SID 100 with the address 1.1.1.1/32. |
| **Step 7** | exit<br>**Example:**<br>`Device(config-router)# exit` | Exits segment routing mode and returns to the configuration terminal mode. |

# Configuring Prefix Attribute N-flag-clear

OSPF advertises prefix SIDs via Extended Prefix TLV in its opaque LSAs. It carries flags for the prefix and one of them is N flag (Node) indicating that any traffic sent along to the prefix is destined to the router originating the LSA. This flag typically marks host routes of router's loopback.

### SUMMARY STEPS

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

### DETAILED STEPS

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

| | Command or Action | Purpose |
|---|---|---|
| | `Device(config-if)# ip ospf prefix-attributes n-flag-clear` | |

# Configuring Explicit Null Attribute With OSPF

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

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

### SUMMARY STEPS

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

### DETAILED STEPS

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

| | Command or Action | Purpose |
|---|---|---|
| | `Device(config-srmpls-attr)# address-family ipv4` | |
| **Step 6** | explicit-null<br>**Example:**<br>`Device(config-srmpls-attr-af)# explicit-null` | Specifies the explicit-null. |
| **Step 7** | exit-address-family<br>**Example:**<br>`Device(config-srmpls-attr-af)# exit-address-family` | Exits the address family. |

# Configuring Segment Routing Label Distribution Protocol Preference With OSPF

### SUMMARY STEPS

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

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device# enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | segment-routing mpls<br>**Example:**<br>`Device(config)# segment-routing mpls` | Configures segment routing mpls mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | set-attributes<br>**Example:**<br>`Device(config-srmpls)# set-attributes` | Sets the attribute. |
| **Step 5** | address-family ipv4<br>**Example:**<br>`Device(config-srmpls-attr)# address-family ipv4` | Specifies the IPv4 address family and enters router address family configuration mode. |
| **Step 6** | sr-label-preferred<br>**Example:**<br>`Device(config-srmpls-attr-af)# sr-label-preferred` | Specifies SR label to be preferred over the LDP. |
| **Step 7** | exit-address-family<br>**Example:**<br>`Device(config-srmpls-attr-af)# exit-address-family` | Exits the address family. |

# Configuring OSPF SRMS

The following command enables the OSPF SRMS and allows OSPF to advertise local mapping entries. OSPF does not send remote entries to the SRMS library. However, OSPF uses the SRMS active policy, which is computed based only on the locally configured mapping entries.

```
[no] segment-routing prefix-sid-map advertise-local
```

# Configuring OSPF SRMS Client

By default, the OSPF SRMS client mode is enabled. OSPF always sends remote prefix-sid-mapping entries received through LSAs, to SRMS. The SRMS active policy is calculated based on both, local and remote mapping entries.

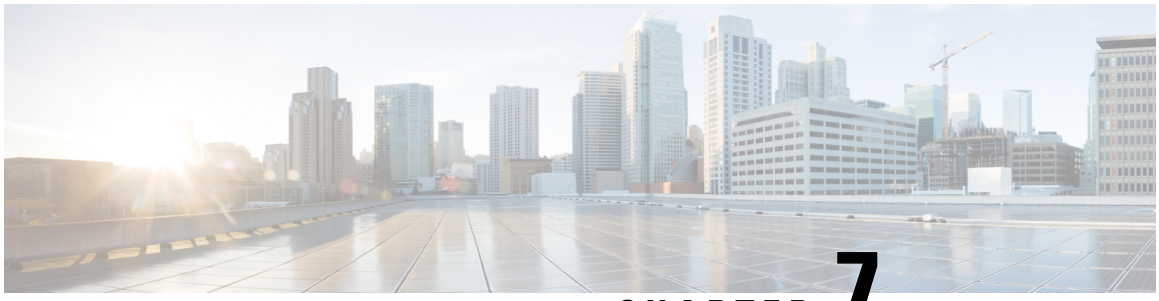
The following command disables the prefix-sid-mapping client functionality and it is configured on the receiver side.

```
segment-routing prefix-sid-map receive [disable]
```

# Additional References for Segment Routing With OSPFv2 Node SID

### Related Documents

| Related Topic | Document Title |
|---|---|
| IP Routing ISIS commands | Cisco IOS IP Routing ISIS commands |



CHAPTER 7

# OSPFv2 Link-protection Topology Independent Loop Free Alternate Fast Reroute

This document describes OSPFv2 implementation of IP Fast Re-Route Feature (IP FRR) using TI -LFA (Topology Independent Loop Free Alternative).


- Feature Information for OSPFv2 Link-protection Topology Independent Loop Free Alternate Fast Reroute, on page 71
- Restrictions for Topology Independent Loop Free Alternate Fast Reroute, on page 72
- Information About OSPFv2 Link-Protection Topology Independent Loop Free Alternate Fast Reroute, on page 72
- How to Configure Topology Independent Loop Free Alternate Fast Reroute, on page 80
- Debugging Topology Independent Loop Free Alternate Fast Reroute, on page 85
- Examples: OSPFv2 Link-Protection Topology Independent Loop Free Alternate Fast Reroute, on page 85


## Feature Information for OSPFv2 Link-protection Topology Independent Loop Free Alternate Fast Reroute

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

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

*Table 6: Feature Information for OSPFv2 Link-protection Topology Independent Loop Free Alternate Fast Reroute*

| Feature Name | Releases | Feature Information |
|---|---|---|
| OSPFv2 Link-protection Topology Independent Loop Free Alternate Fast Reroute | Cisco IOS XE Amsterdam 17.3.2 | Topology-Independent Loop-Free Alternate (TI-LFA) uses segment routing to provide link, node, and Shared Risk Link Groups (SRLG) protection in topologies where other fast reroute techniques cannot provide protection. The goal of TI-LFA is to reduce the packet loss that results while routers converge after a topology change due to a link failure.<br>The following commands were introduced or modified:<br>**fast-reroute per-prefix ti-lfa [area <area> [disable]]**, **fast-reroute per-prefix tie-break node-protecting index <index>**, **fast-reroute per-prefix tie-break node-protecting required index <index>**, **fast-reroute per-prefix tie-break srlg index <index>**, **fast-reroute per-prefix tie-break srlg required index <index>**, **ip ospf fast-reroute per-prefix protection disable**, **ip ospf fast-reroute per-prefix candidate disable**, **show ip ospf fast-reroute ti-lfa tunnels**. |

# Restrictions for Topology Independent Loop Free Alternate Fast Reroute

- TI-LFA is supported only on OSPFv2.
- TI-LFA tunnels are created only if the router supports SR and it is configured with prefix SID. The prefix (or) node SID can be configured as connected SID (or) advertised using the SRMS (Segment Routing Mapping Server).
- TI-LFA is not supported on OSPF point to multi point interfaces.
- TI-LFA does not support Multi Topology Routing (MTR).
- TI-LFA does not create the repair path using virtual link, sham link (or) TE tunnels.
- TI-LFA tunnel is constructed and programmed by explicitly specifying the node (or) set of repair nodes through which the tunnel needs to traverse.

# Information About OSPFv2 Link-Protection Topology Independent Loop Free Alternate Fast Reroute

Topology-Independent Loop-Free Alternate (TI-LFA) uses segment routing to provide link, node, and Shared Risk Link Groups (SRLG) protection in topologies where other fast reroute techniques, such as RLFA (Remote Loop Free Alternative) cannot provide protection. The goal of TI-LFA is to reduce the packet loss that results while routers converge after a topology change due to a link failure. Rapid failure repair (< 50 msec) is achieved

through the use of pre-calculated backup paths that are loop-free and safe to use until the distributed network convergence process is completed.

The following are the major benefits of using TI-LFA:

- Provides 100% coverage for all the prefixes and within 50-msec link and node protection.
- Prevents transient congestion and sub-optimal routing by leveraging on the post-convergence path.
- Protects Label Distribution Protocol (LDP) and IP traffic as well.

# IP Fast Reroute and Remote Loop Free Alternate

IP Fast Reroute (FRR) is a set of techniques that allow rerouting the IP traffic around a failed link or failed node in the network within a very short time (<50ms). One of the techniques that is used is Loop Free Alternates (LFA), which is implemented using OSPF protocol. OSPF currently supports per-prefix directly connected LFA and remote LFA (RLFA). The problem with these LFA algorithms is the topology dependency; the LFA algorithms cannot find a loop-free alternate path through the network for all the topologies.

The per-prefix directly connected LFA (also known as DLFA) provides loop-free alternate path for most triangular topologies, but does not provide good coverage for rectangular or circular topologies. The Remote LFA implementation (RLFA) which uses MPLS forwarding with LDP signaling for tunneling the rerouted traffic to an intermediate node, extends the IPFRR coverage in ring or rectangular topologies. For each link, RLFA defines P-Space (set of nodes reachable from calculating node without crossing the protected link) and Q-Space (set of nodes that can reach the neighbor on the protected link without crossing the protected link itself). The nodes that belong to both P and Q-Spaces are called PQ nodes and can be used as the intermediate node for the protected traffic. RLFA forms targeted LDP session to the PQ node and form the RLFA tunnel. But for the topologies where P and Q-Spaces are disjoint, R-LFA does not provide protection for those prefixes.

# Topology Independent Fast Reroute

Topology Independent Fast Reroute (TI-FRR) is a technique which uses segment routing to provide link protection in any topology assuming the metric on the links in the topology is symmetrical. TI-LFA does not guarantee a backup in the cases where bandwidth on a single link is asymmetrical. TI-LFA only considers loop-free repair paths that are on the post-convergence path. It helps to do better capacity planning of the network.

TI-LFA algorithm allows to create a full explicit path through the network. Using fully specified path may lead to issues in larger topologies due to the number of segments along the path. Specifying the whole path is however not necessary, only a subset of the path is needed to carry the traffic to an intermediate node (release node) which does not loop the traffic back to the protecting node. The TI-LFA algorithm constructs a SR tunnel as the repair path. TI-LFA tunnel is constructed and programmed by explicitly specifying the node (or) set of repair nodes through which the tunnel needs to traverse. The traffic is carried on the tunnel (when the primary path fails) which is also on the post convergence path.

# Topology-Independent Loop Free Alternate

When the local LFA and remote LFA are enabled, there is a good coverage of the prefixes to be protected. However, for some rare topologies that do not have a PQ intersect node, both local and remote LFA will fail to find a release node to protect the failed link. Furthermore, there is no way to prefer a post-convergence path, as the two algorithms have no knowledge of the post-convergence characteristics of the LFA.

To overcome the above limitation, topology-independent LFA (TI-LFA) is supported on an SR-enabled network and provides the following support:

- **Link Protection**—The LFA provides repair path for failure of the link.
- **Local LFA**—Whenever a local LFA on the post convergence path is available, it is preferred over TI-LFA because local LFA does not require additional SID for the repair path. That is, the label for the PQ node is not needed for the release node.
- **Local LFA for extended P space**—For nodes in the extended P space, local LFA is still the most economical method for the repair path. In this case, TI-LFA is not chosen.
- **Tunnel to PQ intersect node**—This is similar to remote LFA except that the repair path is guaranteed on the post convergence path using TI-LFA.
- **Tunnel to PQ disjoint node**—This capability is unique to the TI-LFA in the case when local and remote LFA cannot find a repair path.
- **Tunnel to traverse multiple intersect or disjoint PQ nodes**—TI-LFA provides complete coverage of all prefixes, up to the platform's maximum supported labels.
- **P2P and Broadcast interfaces for the protected link**—TI-LFA protects P2P and broadcast interfaces.
- **Asymmetrical links**—The OSPF metrics between the neighbors are not the same.
- **Multi-homed (anycast) prefix protection**—The same prefix may be originated by multiple nodes and TI-LFA protects the anycast prefixes also by providing post convergence repair path.
- **Protected prefix filtering**—The route-map includes or excludes a list of prefixes to be protected and the option to limit the maximum repair distance to the release node.
- **Tiebreakers**—A subset of existing tiebreakers applicable to TI-LFA is supported.

## Topology Independent Loop Free Alternate Tie-break

Local and remote LFA use default or user-configured heuristics to break the tie when there is more than one path to protect the prefix. The attributes are used to trim down the number of repair paths at the end of the TI-LFA link protection computation before the load balancing.

Local LFA and remote LFA support the following tiebreakers:

- **Linecard-disjoint**—Prefers the line card disjoint repair path.
- **Node-protecting**—Prefers node protecting repair path.
- **SRLG-disjoint**—Prefers SRLG disjoint repair path.
- **Load-sharing**—Distributes repair paths equally among links and prefixes.

When there are two repair paths for a particular prefix, the path that the output port on different line card than that of the primary port is chosen as the repair path.

- **LC-disjoint-index**—If both the repair paths are on the same line card as that of the primary path, then both paths are considered as candidates. If one of the path is on a different line card, then that path is chosen as the repair path.

- **SRLG-disjoint**—Prefers the SRLG disjoint repair path.

The SRLG ID can be configured for each interface. When there are two repair paths for a prefix, the configured SRLG ID for the repair path is compared with that of the primary path SRLG ID. If the SRLG IDs for the secondary path is different than that of the primary, that path is chosen as the repair path.

Effective with Cisco IOS-XE Release 3.18, node-protecting tie-breaker is disabled by default. Tie-breaker default and explicit tie-breaker on the same interface are mutually exclusive. The following tie-breakers are enabled by default on all LFAs:

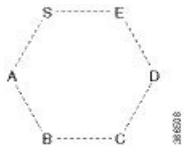
- linecard-disjoint

- lowest-backup-metric
- SRLG-disjoint

# P-Space

The set of routers that can be reached from S on the shortest path tree without traversing S-E is termed the P-space of S with respect to the link S-E.

***Figure 5: A Simple Ring Topology***



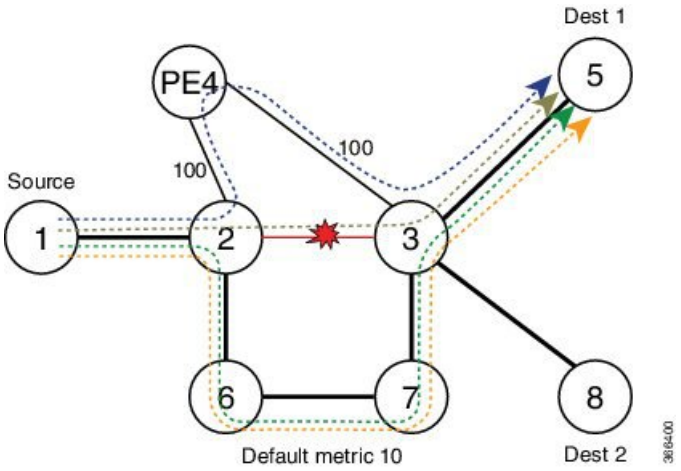

# Q-Space

The set of routers from which the node E can be reached, by normal forwarding without traversing the link S-E, is termed the Q-space of E with respect to the link S-E.

# Post-Convergence Path

Post convergence path is the path that OSPF uses after the link failure. TI-LFA always calculates the repair path which is the post convergence path. You can plan and dimension the post-convergence path to carry the traffic in the case of failure. TI-LFA enforces the post-convergence path by encoding it as a list of segments. The following figure shows an example of TI-LFA using post convergence path:

***Figure 6: TI-LFA Using Post Convergence Path***

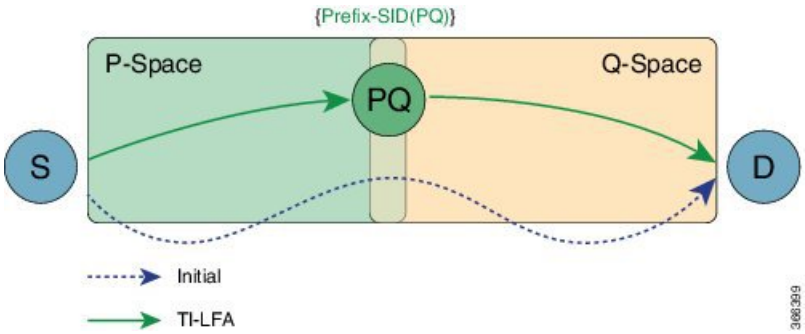



- It protects destination Node 5 on Node 2 against failure of link 2-3.
- Node 2 switches all the traffic destined to Node 5 via core links.

# Per-Destination Link Protection

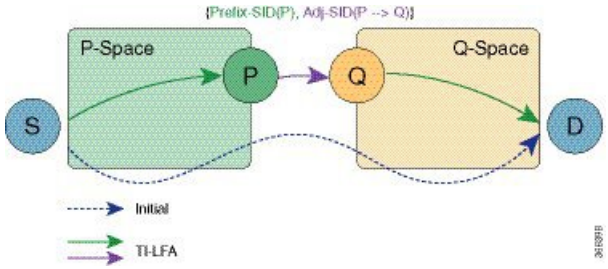
TI-LFA implementation provides per-destination link protection with the number of segments (labels)supported by the underlying hardware. The following figures show the implementation of TI-LFA:

***Figure 7: TI-LFA: { Prefix-SID(PQ) }***




If PQ is a direct neighbor of S, then no additional segment must be pushed.

***Figure 8: TI-LFA: { Prefix-SID(P) , Adj -SID (P -> Q) }***




# Per Interface Loop Free Alternate Enablement

- TI-LFA can be enabled on an area basis.
- TI-LFA backup path is calculated only if TI-LFA protection is enabled on the primary interface which is to be protected. By default all the interfaces are enabled for protection.
- TI-LFA repair path is restricted by the number of labels supported by the hardware. If hardware supports only 2 labels then TI-LFA repair path can protect only those prefixes which can be protected by 2 or lesser segments. For those prefixes which need more than 2 segment remain unprotected.

## Prefix Processing

Once TI-LFA path is calculated for the all the links, prefix processing starts. By default only intra and inter area prefixes are protected. For external prefixes to be protected, you need to enable segment routing globally under the OSPF level.

The primary and repair path should be of the same route type for the prefixes that are protected, that means, if the intra area needs to be protected then the TI-LFA repair path also calculates for the same intra area prefix whether the prefix is unique (or) anycast prefix.

## Anycast Prefix Processing

OSPF TI-LFA also calculates the repair path for the anycast prefixes. Anycast prefixes (or) dual homed prefixes are the prefixes advertised by more than one routers. They could be intra, inter (or), external prefixes. The calculation of TI-LFA repair path for anycast prefixes is as below:

- Assume the prefix P1 is advertised by the routers R1 and R2. The prefix advertised by both the routers should be of the same route type, that is, both R1 and R2 should advertise the prefix as intra area prefix (or inter or external).
- Take the primary path is calculated towards R1 due to the lesser cost.
- When TI-LFA calculates the back up path, it calculates the post convergence path. So, post convergence path need not be towards R1. If the cost to reach R2 (in the post convergence) is shorter, then TI-LFA algorithm chooses the post convergence path towards R2. TI-LFA tunnel is formed towards R2.
- When R2 un-advertises the prefix, then the TI-LFA algorithm is re-calculated towards R1 for the repair path.

## Per-Prefix Loop Free Alternate Tie-Break

IP FRR has the following tie break rules in the order given below. If you have more than one repair path available to choose the best path from, the following tie-break rules are applied. If more than one path matches all the tie break rules, then all the paths are used as repair paths.

- **Post Convergence**: Prefers backup path which is the post convergence path. This is enabled by default and user can not modify this.
- **Primary-path**: Prefers backup path from ECMP set.
- **Interface-disjoint**: Point-to-point interfaces have no alternate next hop for rerouting if the primary gateway fails. You can set the interface-disjoint attribute to prevent selection of such repair paths, thus protecting the interface.
- **Lowest-backup-metric**: Prefers backup path with lowest total metric. This is not applicable for TI-LFA since TI-LFA always chooses the back up path which is lowest cost.
- **LC-disjoint**: Prefers the back up path which is in different line card than that of the primary path.
- **Broadcast-interface-disjoint** : LFA repair paths protect links when a repair path and a protected primary path use different next-hop interfaces. However, on broadcast interfaces if the LFA repair path is computed via the same interface as the primary path and their next-hop gateways are different, in that case the node gets protected, but the link might not be. You can set the broadcast-interface-disjoint attribute to specify that the repair path never crosses the broadcast network the primary path points to, that means, it cannot use the interface and the broadcast network connected to it.
- **Load Sharing**: When more than one repair path matches the above rules, load share the backup paths. This rule also can be modified by the user.

**Note** The user can alter and define the tiebreak rules according to the requirement. In this way, the user can re-prioritize the sequence and/or remove some of the tie break indexes which are not needed.

**Note** The Lowest-backup-metric policy is not applicable for TI-LFA since TI-LFA always chooses the lowest back up path only.

You can see the above rules by using the following command:

```
R2#show ip ospf fast-reroute

            OSPF Router with ID (2.2.2.200) (Process ID 10)

Microloop avoidance is enabled for protected prefixes, delay 5000 msec

Loop-free Fast Reroute protected prefixes:

           Area        Topology name   Priority   Remote LFA Enabled     TI-LFA Enabled
              0                Base        Low                   No                Yes
    AS external                Base        Low                   No                Yes

  Repair path selection policy tiebreaks (built-in default policy):
      0  post-convergence
     10  primary-path
     20  interface-disjoint
     30  lowest-metric
     40  linecard-disjoint
     50  broadcast-interface-disjoint
    256  load-sharing

OSPF/RIB notifications:
 Topology Base: Notification Enabled, Callback Registered

Last SPF calculation started 17:25:51 ago and was running for 3 ms.
```

With the introduction of TI-LFA, the following two tie-break rules are enhanced.

- node-protection
- srlg-protection

The above two tie-break rules are not enabled by default. The user needs to configure the above mentioned tie-break policies.

# Node Protection

TI-LFA node protection provides protection from node failures. Node protecting TI-LFA attempts to calculate the post conversion repair path that protects against the failure of a particular next-hop, not just the link to that particular next-hop.

Node protection is used as a tiebreaker in the implementation of the local LFA also. But when it is combined with TI-LFA, the back up path calculated post convergences with node protecting path. Per-Prefix TI-LFA node protection is disabled by default. The IPFRR TI-LFA node protection features is enabled when the corresponding tiebreak is enabled along with TI-LFA feature, that is,

```
router ospf 10
   [no] fast-reroute per-prefix ti-lfa [area <area> [disable]]
   [no] fast-reroute per-prefix tie-break node-protecting index <index>
   [no] fast-reroute per-prefix tie-break node-protecting required index <index>
```

When you enable node protection, all the other tie break rules also need to manually configured. The node protection is built over the link protection.

The difference between **node-protecting** and **node-protecting required** is in selecting the backup path. When you configure **node-protecting required**, then back up which is chosen has to be the path which does not go through the node (which is part of the link which we are protecting). If no such path is available, then no path is chosen as the backup path.

# Shared Risk Link Groups Protection

A shared risk link group (SRLG) is a group of next-hop interfaces of repair and protected primary paths that have a high likelihood of failing simultaneously. The OSPFv2 Loop-Free Alternate Fast Reroute feature supports only SRLGs that are locally configured on the computing router. With the introduction of TI LFA, the post convergence path which does not share the SRLG group id with the primary path interface will be chosen. In that way, the user will be sure of the SRLG protection whenever the primary link fails.

The IPFRR TI-LFA SRLG protection features is enabled when the corresponding tiebreak is enabled along with Ti-LFA feature, that is,

```
router ospf 10
   [no] fast-reroute per-prefix ti-lfa [area <area> [disable]]
   [no] fast-reroute per-prefix tie-break srlg index <index>
   [no] fast-reroute per-prefix tie-break srlg required index <index>
```

When you enable SRLG protection, you need to manually configure all the other tie break rules. The difference between **srlg-protecting** and **srlg-protecting required** is in selecting the backup path. When you configure **srlg-protecting required**, then back up which is chosen has to be the path which does not share SRLG ID with the primary link which is protected. If no such path is available, then no path is chosen as the backup path.

Whereas, if you configure **srlg-protecting** alone then if the SRLG protection path is not available, the link protection path is chosen as the backup path. And when the SRLG protection path is available, the switchover happens to the SRLG protection path.

# Node-Shared Risk Link Groups Protection

You can configure both node and SRLG protection tie breaks together. This means that the back up path needs to fulfil both the criteria of node protection as well as SRLG protection. In that case, an additional TI-LFA node-SRLG combination protection algorithm is run. The TI-LFA node-SRLG combination algorithm removes the protected node and all members of the interface with the same SRLG group when computing the post-convergence shortest path tree (SPT).

To enable node and SRLG protection tie breaks together, use the following command:

```
router ospf 10
   [no] fast-reroute per-prefix ti-lfa [area <area> [disable]]
   [no] fast-reroute per-prefix tie-break node-protecting index <index>
   [no] fast-reroute per-prefix tie-break srlg index <index>
```

The following show command is used to display the tie break policy:

```
R3#show ip ospf fast-reroute

            OSPF Router with ID (3.3.3.33) (Process ID 10)
```

```
Loop-free Fast Reroute protected prefixes:

           Area        Topology name   Priority   Remote LFA Enabled     TI-LFA Enabled
              0                Base        Low                  No                 No
              1                Base        Low                  No                 No
           1000                Base        Low                  No                 No
    AS external                Base        Low                  No                 No

  Repair path selection policy tiebreaks:
      0  post-convergence
     60  node-protecting
     70  srlg
    256  load-sharing

OSPF/RIB notifications:
 Topology Base: Notification Disabled, Callback Not Registered

Last SPF calculation started 00:00:06 ago and was running for 2 ms.
```

# How to Configure Topology Independent Loop Free Alternate Fast Reroute

## Enabling Topology Independent Loop Free Alternate Fast Reroute

By default, TI-LFA is disabled. You can use protocol enablement to enable TI-LFA.

**Protocol enablement**: Enables TI-LFA in router OSPF mode for all the OSPF areas. Perform the following steps to enable TI-LFA FRR.

```
[no] fast-reroute per-prefix ti-lfa [ area <area> disable]
```

```
router ospf <process>
fast-reroute per-prefix enable area <area> prefix-priority {low | high}
fast-reroute per-prefix ti-lfa [ area <area> disable]
```

You can also use interface command to enable or disable IP FRR on specific interfaces.

```
interface <interface>
ip ospf fast-reroute per-prefix protection disable
ip ospf fast-reroute per-prefix candidate disable
ip ospf fast-reroute per-prefix protection ti-lfa [disable]
```

**Note**

- When TI-LFA is configured on the OSPF router and area wide, area specific configuration takes precedence.
- To protect external prefixes, TI-LFA should be enabled globally.

## Configuring Topology Independent Loop Free Alternate Fast Reroute

This task describes how to enable per-prefix Topology Independent Loop-Free Alternate (TI-LFA) computation to converge traffic flows around link, node, and SRLG failures. TI-LFA can be configured on instance or area

level inherited by lower levels. You can enable or disable per prefix FRR per interface level which is applicable for TI-LFA also.

Before you begin to configure, ensure that the following topology requirements are met:

- Router interfaces are configured as per the topology.
- Routers are configured with OSPF.
- Segment routing is enabled globally as well as under OSPF level.

1. Enables OSPF routing for the specified routing process and enters in router configuration mode.

```
Device(config)# router ospf 10
```

2. Enables FRR.

```
Device(config-router)# fast-reroute per-prefix enable prefix-priority low
```

3. Enables TI-LFA.

```
Device(config-router)# fast-reroute per-prefix ti-lfa
```

4. Enables TI-LFA on the specific area.

```
Device(config-router)# fast-reroute per-prefix ti-lfa area 0
```

5. Exits the TI-LFA mode.

```
Device(config-router)# exit
```

6. Enters the interface mode.

```
Device(config)#interface ethernet 0/0
```

7. If you do not wish to enable FRR on a specific inteface, use the protection disable command.

```
Device(config-if)#ip ospf fast-reroute per-prefix protection disable
```

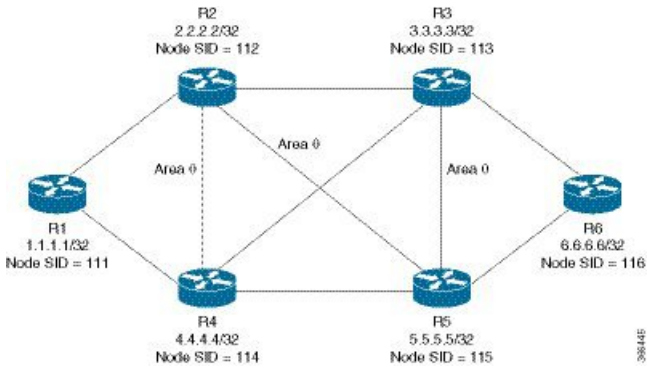
8. If you do not wish a specific interface to be enabled as a repair path, use the candidate disable command.

```
Device(config-if)#ip ospf fast-reroute per-prefix candidate disable
```

# Configuring Topology Independent Fast Reroute Tie-breaker

You need to enable segment routing on all the routers with prefix SIDs configured for all the nodes. Use the following topology as a reference to understand the configuration.

***Figure 9: Configuration Example***




Let us take the device R2 which is protecting the link between R2 and R3. The configuration at R2:

```
router ospf 10
fast-reroute per-prefix enable prefix-priority low
fast-reroute per-prefix ti-lfa
segment-routing mpls
segment-routing area 0 mpls
fast-reroute per-prefix enable prefix-priority low
fast-reroute per-prefix ti-lfa
fast-reroute per-prefix ti-lfa area 0
fast-reroute per-prefix tie-break node-protecting index 60
fast-reroute per-prefix tie-break srlg index 70
mpls traffic-eng router-id Loopback1
mpls traffic-eng area 0

interface GigabitEthernet4   //interface connecting to the router 4
ip address 100.101.4.4 255.255.255.0
ip ospf 10 area 0
ip ospf network point-to-point
srlg gid 10
negotiation auto

interface GigabitEthernet3  //interface connecting to the router 3
ip address 100.101.3.3 255.255.255.0
ip ospf 10 area 0
ip ospf network point-to-point
srlg gid 10
negotiation auto

interface GigabitEthernet5   //interface connecting to the router 2
ip address 100.101.5.5 255.255.255.0
ip ospf 10 area 0
ip ospf network point-to-point
srlg gid 20
negotiation auto



interface loopback2
ip address 2.2.2.2/32
ip ospf 10 area 0
```

**Note** In all the other devices, configuration of segment routing and assignment of connected prefix SIDs need to be done.

**How Node Protection Works**: Using the same topology as an example, let us take the case where you are protecting the link between R2 and R3 and also the prefix which is leant from R6. In that case, let us assume that the primary path for the prefix is via R2-R3. So, our primary path is R2---R3---R6 and we are protecting the link R2---R3.

In this scenario, only link-protection is configured and enabled. When you enable TI-LFA under OSPF process, then you get the following paths provided the cost for all the paths are equal:

R2----R4----R5---R6

R2---R5----R3---R6

R2----R5---R6

If you have only link protection configured, then all the three paths will be chosen and they will share the load amongst them.

If you wish to configure node protection, then the backup would be calculated in such a way that the back up path does not contain the node that you are protecting. In this example, the node R3 in the back up is not required. As a result, only the following two paths would be chosen as the back up paths:

R2----R4----R5---R6

R2----R5---R6

It is possible that R2---R5---R3---R6 have the lesser cost than the above two paths. But since the node protection is configured, only the paths amongst the above two will be considered.

**How SRLG Protection Works**: SRLG protection further eliminates the back up paths in a such a way that the primary path and the backup does not share the same SRLG ID. Suppose the following back up paths are available:

R2----R4----R5---R6

R2----R5---R6

Then, the SRLG ID of (R2----R4) and (R2----R5) are compared against the primary interface (R2----R3) which is 10. It is noticed that only the interface R2----R5 has different SRLG ID which is 20. So, only the backup path R2---R5---R6 will be chosen.

# Verifying Topology Independent Fast Reroute Tunnels

You can use the following command, to check the TI LFA tunnels:

```
Device#show ip ospf fast-reroute ti-lfa tunnels


            OSPF Router with ID (2.2.2.200) (Process ID 10)

                        Area with ID (0)

                Base Topology (MTID 0)
```

```
Tunnel                Interface         Next Hop          Mid/End Point    Label
-------------------------------------------------------------------------------
MPLS-SR-Tunnel2       Et1/1             2.7.0.7           1.1.1.1          16020
MPLS-SR-Tunnel6       Et0/3             2.8.0.0           3.3.3.3          16003
MPLS-SR-Tunnel7       Et1/1             2.7.0.7           1.1.1.1          16020
                                                          5.5.5.5          16005
                                                          3.3.3.3          16003
MPLS-SR-Tunnel5       Et0/3             2.8.0.0           5.5.5.5          16005
MPLS-SR-Tunnel1       Et1/1             2.7.0.7           1.1.1.1          16020
                                                          5.5.5.5          16005
MPLS-SR-Tunnel3       Et1/1             2.7.0.7           6.6.6.6          16006
```

You can use the following command, to check the route in OSPF routing table with primary and repair path:

```
Device#show ip ospf rib 6.6.6.6


            OSPF Router with ID (2.2.2.200) (Process ID 10)


                Base Topology (MTID 0)

OSPF local RIB
Codes: * - Best, > - Installed in global RIB
LSA: type/LSID/originator

*>  6.6.6.6/32, Intra, cost 31, area 0
     SPF Instance 19, age 02:12:11
      contributing LSA: 10/7.0.0.0/6.6.6.6 (area 0)
     SID: 6
     CSTR Local label: 0
     Properties: Sid, LblRegd, SidIndex, N-Flag, TeAnn
     Flags: RIB, HiPrio
      via 2.7.0.7, Ethernet1/1 label 16006
       Flags: RIB
       LSA: 1/6.6.6.6/6.6.6.6
      PostConvrg repair path via 3.3.3.3, MPLS-SR-Tunnel6 label 16006, cost 81, Lbl cnt 1
       Flags: RIB, Repair, PostConvrg, IntfDj, LC Dj
       LSA: 1/6.6.6.6/6.6.6.6
```

You can use the following command, to display the route in the IP routing table:

```
Device#show ip route 6.6.6.6
Routing entry for 6.6.6.6/32
  Known via "ospf 10", distance 110, metric 31, type intra area
  Last update from 2.7.0.7 on Ethernet1/1, 00:25:14 ago
 SR Incoming Label: 16006
  Routing Descriptor Blocks:
  * 2.7.0.7, from 6.6.6.6, 00:25:14 ago, via Ethernet1/1, merge-labels
      Route metric is 31, traffic share count is 1
      MPLS label: 16006
      MPLS Flags: NSF
      Repair Path: 3.3.3.3, via MPLS-SR-Tunnel6
```

# Debugging Topology Independent Loop Free Alternate Fast Reroute

You can use the following commands to debug TI-LFA FRR:

```
debug ip ospf fast-reroute spf
debug ip ospf fast-reroute spf detail
debug ip ospf fast-reroute rib
debug ip ospf fast-reroute rib [<access-list>]
```

# Examples: OSPFv2 Link-Protection Topology Independent Loop Free Alternate Fast Reroute

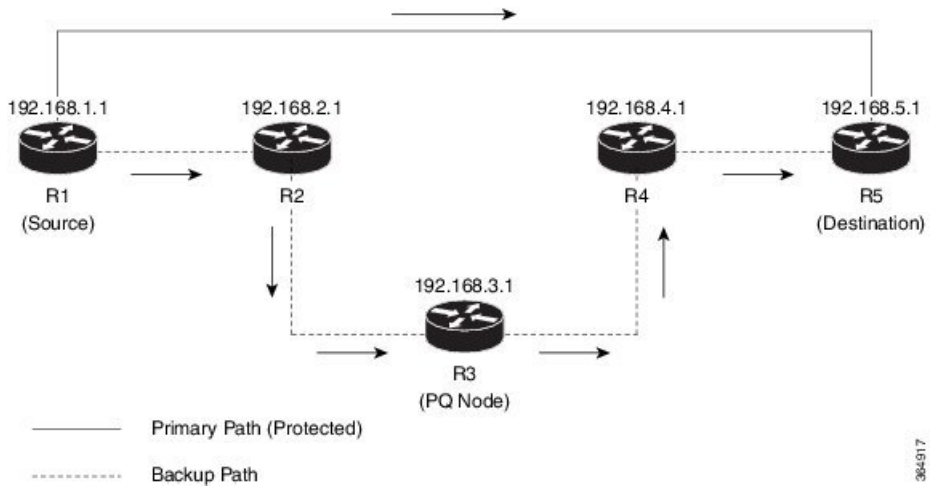
The following are the examples for the OSPFv2 Link-Protection TI-LFA FRR.

## Example: Configuring Topology Independent Loop Free Alternate Fast Reroute

This example shows how to configure TI-LFA for segment routing TE tunnels using single or disjoint PQ nodes. The following are the two topologies used:

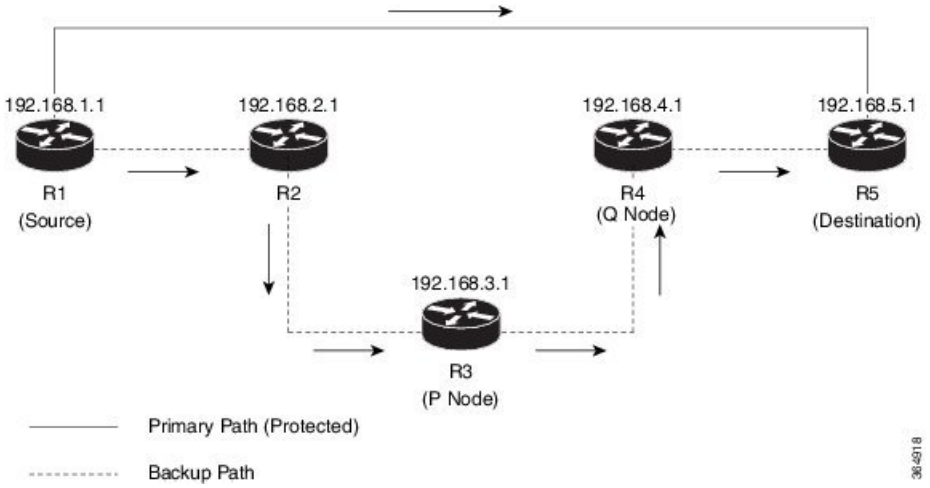
- Topology 1: A single PQ Node and therefore has two SIDs from the source router, R1 through the PQ Node to the destination router, R5.

***Figure 10: Topology 1: Single PQ Node***



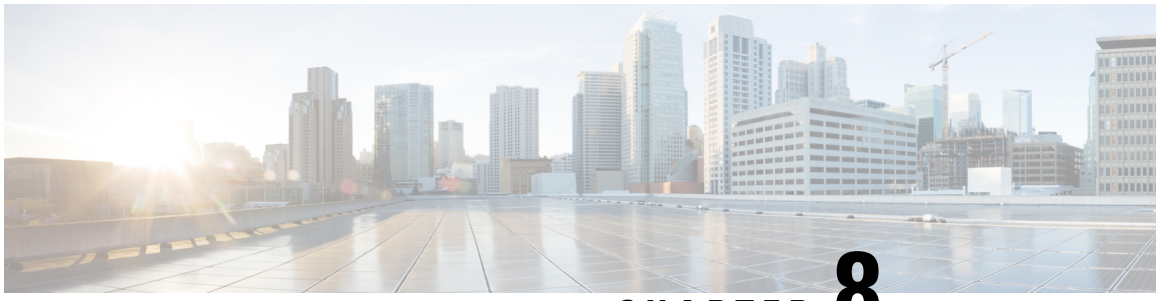

- Topology 2: Disjoint PQ Nodes and therefore consists of three SIDs from the source router R1, through the P Node and the Q Node to the destination router, R5.

*Figure 11: Topology 2: Disjoint PQ Nodes*




Configure TI-LFA for OSPF on the source router (R1) interface connecting to the destination router (R5).

```
Device(config)# router ospf 10
Device(config-router)# fast-reroute per-prefix enable prefix-priority low
Device(config-router)# fast-reroute per-prefix ti-lfa
Device(config-router)# fast-reroute per-prefix ti-lfa area 0
Device(config-router)# exit
```



CHAPTER 8

# Segment Routing Traffic Engineering With OSPF

This chapter describes how Segment Routing traffic engineering can be implemented using OSPF.


- Feature Information for Segment Routing Traffic Engineering With OSPF, on page 87
- Restrictions for Segment Routing Traffic Engineering With OSPF, on page 88
- Information About Segment Routing Traffic Engineering With OSPF, on page 88
- How to Configure Segment Routing Traffic Engineering With OSPF, on page 96
- Verifying Configuration of the SR-TE Tunnels, on page 104


## Feature Information for Segment Routing Traffic Engineering With OSPF

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

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

***Table 7: Feature Information for Segment Routing Traffic Engineering With OSPF***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing Traffic Engineering With OSPF | Cisco IOS XE Amsterdam 17.3.2 | A Traffic Engineered (TE) tunnel is a container of TE LSP(s) instantiated between the tunnel ingress and the tunnel destination. A TE tunnel may instantiate one or more SR-TE LSP(s) that are associated with the same tunnel.<br>The following commands were added or modified:<br>**show mpls traffic-eng tunnels**, **tunnel mpls traffic-eng path-option 10 dynamic segment-routing**, **tunnel mpls traffic-eng path-option 10 segment-routing**, **tunnel mpls traffic-eng path-option 10 explicit name MIXED_PATH segment-routingtunnel mpls traffic-eng path-option 10 explicit name IP_PATH1 segment-routingtunnel mpls traffic-eng path-option 10 explicit name NODE_PATH segment-routing**. |

# Restrictions for Segment Routing Traffic Engineering With OSPF

- Segment Routing Traffic Engineering is supported only on OSPFv2.
- SR-TE is not supported on broadcast interfaces; it is supported only point-to-point interfaces.
- Only one instance of protocol should be enabled for TE at a given point of time.

# Information About Segment Routing Traffic Engineering With OSPF

A Traffic Engineered (TE) tunnel is a container of TE LSP(s) instantiated between the tunnel ingress and the tunnel destination. A TE tunnel may instantiate one or more SR-TE LSP(s) that are associated with the same tunnel. The SR-TE LSP path may not necessarily follow the same IGP path to a destination node. In this case, the SR-TE path can be specified a set of prefix-SID(s) and/or adjacency-SID(s) of nodes and/or links to be traversed by the SR-TE LSP.

The head-end imposes the corresponding MPLS label stack on to outgoing packets to be carried over the tunnel. Each transit node along the SR-TE LSP path uses the incoming top label to select the next-hop, pop or swap the label, and forward the packet to the next node with the remainder of the label stack, until the packet reaches the ultimate destination. OSPF provides TE with the topology and SR related information. SR related information include SRGB/prefix/Adjacency SIDs of all nodes/links with SR enabled in the network.

## Benefits of Using Segment Routing Traffic Engineering With OSPF

Segment routing traffic engineering offers a comprehensive support for all useful optimizations and constraints, for example:

- Latency
- Bandwidth
- Disjointness
- Resource avoidance

OSPFv2 provides the following functionalities for SR-TE:

- OSPFv2 provides SR information along with TE topology information to TE module.
- TE uses this information to construct SR TE path/tunnel comprising of one or more segments - with the combination of prefix and/or adjacency segments.
- For the prefixes TE is interested in, OSPF provides first hop resolution to setup the forwarding plane.
- SR TE tunnels are also advertised back into OSPF (like RSVP TE tunnels) for diverting traffic over the SR-TE tunnels.

# OSPFv2 Segment Routing Traffic Engineering Functionalities

OSPFv2 perform the following functionalities for SR-TE:

- OSPFv2 provides SR information along with TE topology information to TE module.
- TE uses this information to construct SR TE path/tunnel comprising of one or more segments - with the combination of prefix and/or adjacency segments.
- For the prefixes TE is interested in, OSPF provides first hop resolution to setup the forwarding plane.
- SR TE tunnels are also advertised back into OSPF (like RSVP TE tunnels) for diverting traffic over the SR-TE tunnels.

# Protected Adjacency SID

Segment routing creates protected adjacency SID for point to point to point interfaces and broadcast interfaces. It advertises them to the extended link-state advertisement (LSA) along with the unprotected adjacency SID. Protected adjacency SID can have a repair path, but it is not guaranteed to have a repair path.

# Traffic Engineering Interfaces

In order to support SR-TE functionality, TE interfaces with various components, and with IGP (OSPF and ISIS) to distribute and receive information on TE topology. For SR-TE support, OSPF needs to additionally provide SR information to TE that it had received through various LSAs, for example,

- Router Information LSA
- Extended Prefix LSA
- Extended Link LSA

TE interfaces distribute information, such as bandwidth resources, constraints, capabilities, and other attributes, associated with the links that are configured for TE. The link information is distributed to other routers using opaque LSAs and is used by TE to create a local topology database. The topology database is a key element in allowing TE to compute a suitable constraint-based path for establishing an LSP. TE also interfaces with the IGP to notify when a TE headend interface can be considered for routing packets.

# Unnumbered Support

IS-IS description of an unnumbered link does not contain remote interface ID information. The remote interface ID of an unnumbered link is required to include the unnumbered link as part of the SR-TE tunnel.

# Segment Routing Traffic Engineering Support for Forwarding Adjacency

MPLS TE forwarding adjacency feature is supported by OSPF. In this, TE tunnel is considered as a link in the IGP network. TE tunnel interfaces are advertised in the IGP network like any other links. Routers can then use these links to compute the shortest path tree (SPT).

**Note** This feature is not supported with the SR-TE tunnels.

# Segment Routing Traffic Engineering Support for Auto-route Announce

MPLS TE auto-route announce feature is supported by OSPF, that uses TE Tunnel as the first-hop, if the node is reachable via that tunnel. It allows the traffic to the nodes that are downstream to the tail-end of the TE tunnel flows through the tunnel. OSPF supports auto-route over the SR-TE tunnels similar to the MPLS TE tunnels setup using RSVP.

The TE tunnel that instantiates an SR-TE LSP can be Auto-route Announced (AA) into IGP (OSPF and ISIS) as an IGP shortcut. The IGP uses the TE tunnel as next hop and installs routes in RIB for all IP prefixes whose shortest path falls behind the TE tunnel destination. Auto-route announce for of TE tunnels is supported to carry IPV4 prefixes.

## Auto-route Announce IP2MPLS

The auto-routeIP2MPLS feature is introduced for SR tunnels to avoid potential packet from looping indefinitely between the SR-TE tunnel headend/ingress and a node that is pointing/routing the packet back to the headend/ingress.

The solution consists in the headend programming in forwarding two sets of path(s) for the prefixes that are mapped over the SR-TE tunnel. The first is the pure IP route for the prefix(es) mapped on the and having the outgoing interface as the tunnel interface. This allows mapping IP traffic directly over the tunnel. The second is the MPLS path for the prefixes mapped on the tunnel. For this the prefix-SID label is programmed with the IGP shortest path outgoing interface(s), that is, non tunnel output interfaces.

# SR-TE LSP Instantiation

A Traffic Engineered (TE) tunnel is a container of one or more instantiated TE LSPs. An SR-TE LSP is instantiated by configuring ‘segment-routing’ on the path-option of the TE tunnel. The traffic mapped to the tunnel is forwarded over the primary SR-TE instantiated LSP.

Multiple path-options can also be configured under the same tunnel. Each path-option is assigned a preference index or a path-option index that is used to determine the more favorable path-option for instantiating the primary LSP—the lower the path-option preference index, the more favorable the path-option. The other less favorable path-options under the same TE tunnel are considered secondary path-options and may be used once the currently used path-option is invalidated (for example, due to a failure on the path.

**Note** A forwarding state is maintained for the primary LSP only.

# Tunnel Path Affinity Validation

The affinity of a tunnel path can be specified using the command **tunnel mpls traffic-eng affinity** under the tunnel interface.

The head-end validates that the specified SR path is compliant with the configured affinity. This necessitates that the paths of each segment of the SR path be validated against the specified constraint. The path is declared invalid against the configured affinity constraints if at least a single segment of the path does not satisfy the configured affinity.

# SR-TE Traffic Load Balancing

SR-TE tunnels support the following load-balancing options:

## Load Balancing on Port Channel TE Links

Port Channel interfaces carry the SR-TE LSP traffic. This traffic load balances over port channel member links as well as over bundle interfaces on the head or mid of an SR-TE LSP.

## Load Balancing on Single Tunnel

While using the equal cost multi path protocol (ECMP), the path to a specific prefix-SID may point to multiple next-hops. And if the SR-TE LSP path traverses one or more prefix-SIDs that have ECMP, the SR-TE LSP traffic load-balances on the ECMP paths of each traversed prefix-SID from the head-end or any midpoint traversed node along the SR-TE LSP path.

## Load Balancing on Multiple Tunnels

Multiple TE tunnels can be used as next-hop paths for routes to specific IP prefixes either by configuring static route on multiple tunnels, or auto-route announcing multiple parallel tunnels to the same destination. In such cases, the tunnels share the traffic load equally or load balance traffic on multiple parallel tunnels. It is also possible to allow Unequal Load Balance (UELB) with an explicit per tunnel configuration at the tunnel head-end. In this case, the tunnel load-share is passed from MPLS-TE to forwarding plane.

The tunnel load-share feature continues to work for TE tunnels that instantiate the SR-TE LSPs.

# SR-TE Tunnel Re-optimization

TE tunnel re-optimization occurs when the head-end determines that there is a more optimal path available than the one currently used. For example, in case of a failure along the SR-TE LSP path, the head-end could detect and revert to a more optimal path by triggering re-optimization.

Tunnels that instantiate SR-TE LSP can re-optimize without affecting the traffic carried over the tunnel.

Re-optimization can occur because:

- the explicit path hops used by the primary SR-TE LSP explicit path are modified,
- the head-end determines the currently used path-option are invalid due to either a topology path disconnect, or a missing SID in the SID database that is specified in the explicit-path
- a more favorable path-option (lower index) becomes available

When the head-end detects a failure on a protected SR adjacency-SID that is traversed by an SR-TE LSP, it starts the invalidation timer. If the timer expires and the head-end is still using the failed path because it is unable to reroute on a different path, the tunnel state is brought ‘down’ to avoid a null route from being sent along with the traffic. Once the tunnel is down, services on the tunnel converge to take a different path.

The following is a sample output of a manual re-optimization example. In this example, the path-option is changed from ‘10’ to ‘20’.

```
Router# mpls traffic-eng reoptimize tunnel 1 path-option 20
The targeted path-option is not in lock down mode. Continue? [no]: yes
Router# show mpls traffic-eng tunnels tunnel1
Name: R1_t1                           (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 20, (SEGMENT-ROUTING) type explicit IP_PATH (Basis for Setup)
    path option 10, (SEGMENT-ROUTING) type dynamic
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 20 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours, 9 minutes
      Time since path change: 14 seconds
      Number of LSP IDs (Tun_Instances) used: 1819
    Current LSP: [ID: 1819]
      Uptime: 17 seconds
      Selection: reoptimization
    Prior LSP: [ID: 1818]
      ID: path option unknown
      Removal Trigger: reoptimization completed
  Tun_Instance: 1819
  Segment-Routing Path Info (isis level-1)
    Segment0[Node]: 4.4.4.4, Label: 114
    Segment1[Node]: 5.5.5.5, Label: 115
    Segment2[Node]: 6.6.6.6, Label: 116
```

## SR-TE With Lockdown Option

The **lockdown** option prevents SR-TE from re-optimizing to a better path. However, it does not prevent signaling the existence of a new path.

```
interface Tunnel1
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng path-option 10 segment-routing lockdown
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10                      (Tunnel1) Destination: 6.6.6.6

  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (LOCKDOWN) type segment-routing (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
```

```
  AutoRoute: enabled  LockDown: enabled  Loadshare: 10 [200000000]
  auto-bw: disabled
  Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
Active Path Option Parameters:
  State: segment-routing path option 10 is active
  BandwidthOverride: disabled  LockDown: enabled   Verbatim: disabled
History:
  Tunnel:
    Time since created: 6 days, 19 hours, 22 minutes
    Time since path change: 1 minutes, 26 seconds
    Number of LSP IDs (Tun_Instances) used: 1822
  Current LSP: [ID: 1822]
    Uptime: 1 minutes, 26 seconds
    Selection: reoptimization
  Prior LSP: [ID: 1821]
    ID: path option unknown
    Removal Trigger: configuration changed
Tun_Instance: 1822
Segment-Routing Path Info (isis  level-1)
  Segment0[Node]: 6.6.6.6, Label: 116
```

# SR-TE Tunnel Protection

Protection for SR TE tunnels can take any of the following alternatives:

## IP-FRR Local Repair Protection

On an SR-TE LSP head-end or mid-point node, IP-FRR is used to compute and program the backup protection path for the prefix-SID or adjacency-SID label.

With IP-FRR, backup repair paths are pre-computed and pre-programmed by IGPs *before* a link or node failure. The failure of a link triggers its immediate withdrawal from the TE topology (link advertisement withdrawal). This allows the head-end to detect the failure of an SR-TE LSP traversing the failed adjacency-SID.

When a protected adjacency-SID fails, the failed adjacency-SID label and associated forwarding are kept functional for a specified period of time (5 to 15 minutes) to allow all SR TE tunnel head-ends to detect and react to the failure. Traffic using the adjacency-SID label continues to be FRR protected even if there are subsequent topology updates that change the backup repair path. In this case, the IGPs update the backup repair path while FRR is active to reroute traffic on the newly-computed backup path.

When the primary path of a protected prefix-SID fails, the PLR reroutes to the backup path. The head-end remains transparent to the failure and continues to use the SR-TE LSP as a valid path.

IP-FRR provides protection for adjacency and prefix-SIDs against link failures only.

## Tunnel Path Protection

Path protection is the instantiation of one or more standby LSPs to protect against the failure of the primary LSP of a single TE tunnel.

Path protection protects against failures by pre-computing and pre-provisioning secondary paths that are failure diverse with the primary path-option under the same tunnel. This protection is achieved by computing a path that excludes prefix-SIDs and adjacency-SIDs traversed by the primary LSP or by computing a path that excludes SRLGs of the primary SR-TE LSP path.

In the event of a failure of the primary SR-TE LSP, at least one standby SR-TE LSP is used for the tunnel. Multiple secondary path-options can be configured to be used as standby SR-TE LSPs paths.

# SR-TE LSP Path Verification

SR-TE tunnel functionality requires that the head-end perform initial verification of the tunnel path as well as the subsequent tracking of the reachability of the tunnel tail-end and traversed segments.

Path verification for SR-TE LSP paths is triggered whenever MPLS-TE is notified of any topology changes or SR SID updates.

The SR-TE LSP validation steps consist of the following checks:

## Topology Path Validation

The head-end validates the path of an SR-TE LSP for connectivity against the TE topology. MPLS-TE head-end checks if links corresponding to the adjacency SIDs are connected in the TE topology.

For newly-instantiated SR-TE LSPs, if the head-end detects a discontinuity on any link of the SR-TE path, that path is considered invalid and is not used. If the tunnel has other path-options with valid paths, those paths are used to instantiate the tunnel LSP.

For TE tunnels with existing instantiated SR-TE LSP, if the head-end detects a discontinuity on any link, the head-end assumes a fault has occurred on that link. In this case, the local repair protection, such as the IP FRR, come in to effect. The IGPs continue to sustain the protected adjacency label and associated forwarding after the adjacency is lost for some time. This allows the head-ends enough time to reroute the tunnels onto different paths that are not affected by the same failure. The head-end starts a tunnel invalidation timer once it detects the link failure to attempt to reroute the tunnel onto other available path-options with valid paths.

If the TE tunnel is configured with other path-options that are not affected by the failure and are validated, the head-end uses one of those path-options to reroute (and re-optimize) the tunnel by instantiating a new primary LSP for the tunnel using the unaffected path.

If no other valid path-options exist under the same tunnel, or if the TE tunnel is configured with only one path-option that is affected by the failure, the head-end starts an invalidation timer after which it brings the tunnel state to ‘down’. This action avoids a null route from being sent along with traffic flowing over the affected SR-TE LSP, and allows services riding over the tunnel to reroute over different available paths at the head-end. There is an invalidation drop configuration that keeps the tunnel 'up', but drops the traffic when the invalidation timer expires.

For intra-area SR-TE LSPs, the head-end has full visibility over the LSP path, and validates the path to the ultimate LSP destination. However, for inter-area LSPs, the head-end has partial visibility over the LSP path—only up to the first ABR. In this case, the head-end can only validate the path from the ingress to the first ABR. Any failure along the LSP beyond the first ABR node is invisible to the head-end, and other mechanisms to detect such failures, such as BFD over LSP are assumed.

## SR SID Validation

SID hops of an SR-TE LSP are used to determine the outgoing MPLS label stack to be imposed on the outgoing packets carried over the SR-TE LSP of a TE tunnel. A database of global and local adjacency-SIDs is populated from the information received from IGPs and maintained in MPLS-TE. Using a SID that is not available in the MPLS TE database invalidates the path-option using the explicit-path. The path-option, in this case, is not used to instantiate the SR TE LSP. Also, withdrawing, adding, or modifying a SID in the MPLS-TE SID-database, results in the MPLS-TE head-end verifying all tunnels with SR path-options (in-use or secondary) and invokes proper handling.

## LSP Egress Interface

When the SR-TE LSP uses an adjacency-SID for the first path hop, TE monitors the interface state and IGP adjacency state associated with the adjacency-SID and the node that the SR-TE LSP egresses on. If the interface or adjacency goes down, TE can assume a fault occurred on the SR-TE LSP path and take the same reactive actions described in the previous sections.

**Note** When the SR-TE LSP uses a prefix-SID for the first hop, TE cannot directly infer on which interface the tunnel egresses. TE relies on the IP reachability information of the prefix to determine if connectivity to the first hop is maintained.

## IP Reachability Validation

MPLS-TE validates that the nodes corresponding to the prefix-SIDs are IP reachable before declaring the SR path valid. MPLS-TE detects path changes for the IP prefixes corresponding to the adjacency or prefix SIDs of the SR-TE LSP path. If the node announcing a specific SID loses IP reachability. due to a link or node failure, MPLS-TE is notified of the path change (no path). MPLS-TE reacts by invalidating the current SR-TE LSP path, and may use other path-options with a valid path, if any to instantiate a new SR-TE LSP.

**Note** Since IP-FRR does not offer protection against failure of a node that is being traversed by an SR-TE LSP (such as, a prefix-SID failure along the SR-TE LSP path), the head-end immediately reacts to IP route reachability loss for prefix-SID node by setting the tunnel state to ‘down’ and removes the tunnel forwarding entry if there are no other path-options with valid path for the affected tunnel.

## Tunnel Path Resource Avoidance Validation

You can specify a set of addresses to be validated as excluded from being traversed by SR-TE tunnel packets. To achieve this, the head-end runs the per-segment verification checks and validates that the specified node, prefix or link addresses are indeed excluded from the tunnel in the SR path. The tunnel resource avoidance checks can be enabled per path using the commands below. The list of addresses to be excluded are defined and the name of the list is referenced in the path-option.

```
interface tunnel100
 tunnel mpls traffic-eng path-option 1 explicit name EXCLUDE segment-routing
ip explicit-path name EXCLUDE enable
 exclude-address 192.168.0.2
 exclude-address 192.168.0.4
 exclude-address 192.168.0.3
!
```

## SR-TE LSP Explicit Null

MPLS-TE tunnel head-end does not impose explicit-null at the bottom of the stack. When penultimate hop popping (PHP) is enabled for SR prefix SIDs or when an adjacency SID is the last hop of the SR-TE LSP, the packet may arrive at the tail-end without a transport label. However, in some cases, it is desirable that the packet arrive at the tail-end with explicit-null label, and in such case, the head-end will impose an explicit-null label at the top of the label stack.

### Verbatim Path Support

MPLS TE LSPs usually require that all the nodes in the network are TE aware which means that they have IGP extensions to TE in place. However, some network administrators want the ability to build TE LSPs to traverse nodes that do not support IGP extensions to TE, but that do support RSVP extensions to TE. Verbatim LSPs are helpful when all or some of the intermediate nodes in a network do not support IGP extensions for TE.

When this feature is enabled, the IP explicit path is not checked against the TE topology database. Since the TE topology database is not verified, a Path message with IP explicit path information is routed using the shortest path first (SPF) algorithm for IP routing.

# How to Configure Segment Routing Traffic Engineering With OSPF

Perform the following steps to configure Segment Routing Traffic Engineering With OSPF.

## Enabling Segment Routing Traffic Engineering With OSPF

OSPF Segment Routing traffic engineering is enabled when the segment-routing is enabled along with mpls traffic engineering. SR-TE support is turned on in an area when you enable SR & MPLS TE in that area.

```
router ospf 10
 router-id 10.10.10.2
 segment-routing mpls
  mpls traffic-eng area 0
```

## Configuring Path Option for a TE Tunnel

The **segment-routing** keyword indicates that the specified path is programmed as an SR path:

```
Device(config)# interface tunnel 100
Device(config-if)# tunnel mpls traffic-eng path-option 1 explicit name foo segment-routing
Device(config-if)# tunnel mpls traffic-eng path-option 2 dynamic segment-routing
Device(config-if)# tunnel mpls traffic-eng path-option 3 segment-routing
```

**Note** With IP unnumbered interfaces dynamic path is not supported.

When the path-option type for an operational SR tunnel is changed from SR to non-SR (for example, **dynamic**), the existing forwarding entry of the tunnel is deleted.

Segment Routing can be enabled or disabled on an existing secondary or an in-use path-option. If the tunnel uses a signaled RSVP-TE explicit path-option and segment routing is enabled on that tunnel, the RSVP-TE LSP is torn, and the SR-TE LSP is instantiated using the same path-option. Conversely, if segment routing is disabled on a path-option that is in use by the primary LSP, the tunnel goes down intermittently and a new RSVP-TE LSP will be signaled using the same explicit path.

If the segment-routing path-option is enabled on a secondary path-option (that is, not in-use by the tunnel's primary LSP), the tunnel is checked to evaluate if the newly specified SR-TE LSP path-option is valid and more favorable to use for the tunnel primary LSP.

# Configuring SR Explicit Path Hops

The following SR-TE explicit path hops are supported:

- IP addresses
- MPLS labels
- Mix of IP addresses and MPLS labels

For intra-area LSPs, the explicit path can be specified as a list of IP addresses.

```
Device(config)# ip explicit-path name foo
Device(config-ip-expl-path)# index 10 next-address 1.1.1.1 node address
Device(config-ip-expl-path)# index 20 next-address 12.12.12.2 link address
```

**Note** When using IP unnumbered interfaces, you cannot specify next hop address as an explicit path index. It should be node address or label.

The explicit path can also be specified as segment-routing SIDs:

```
Device(config)# ip explicit-path name foo
Device(config-ip-expl-path)# index 10 next-label 20
```

**Note** IP addresses cannot be used after using the label in MIXED_PATH.

# Configuring Tunnel Path Affinity Validation

The affinity of a tunnel path can be specified using the command **tunnel mpls traffic-eng affinity** under the tunnel interface.

The head-end validates that the specified SR path is compliant with the configured affinity. This necessitates that the paths of each segment of the SR path be validated against the specified constraint. The path is declared invalid against the configured affinity constraints if at least a single segment of the path does not satisfy the configured affinity.

```
interface Tunnel1
 no ip address
 tunnel mode mpls traffic-eng
 tunnel destination 5.5.5.5
 tunnel mpls traffic-eng priority 5 5
 tunnel mpls traffic-eng bandwidth 100
 tunnel mpls traffic-eng affinity 0x1 mask 0xFFFF
        tunnel mpls traffic-eng path-option 10 dynamic segment-routing
Router# show tunnel ??
Name: R1_t1                             (Tunnel1) Destination: 5.5.5.5
  Status:
```

```
  Admin: up         Oper: up     Path: valid       Signalling: connected
  path option 10, (SEGMENT-ROUTING) type dynamic (Basis for Setup, path weight 20)
Config Parameters:
  Bandwidth: 100      kbps (Global)  Priority: 5  5   Affinity: 0x1/0xFFFF
  Metric Type: TE (default)
  Path Selection:
   Protection: any (default)
  Path-selection Tiebreaker:
    Global: not set   Tunnel Specific: not set   Effective: min-fill (default)
  Hop Limit: disabled
  Cost Limit: disabled
  Path-invalidation timeout: 10000 msec (default), Action: Tear
  AutoRoute: disabled LockDown: disabled Loadshare: 100 [0] bw-based
  auto-bw: disabled
  Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
Active Path Option Parameters:
  State: dynamic path option 10 is active
  BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
Node Hop Count: 2
History:
  Tunnel:
    Time since created: 10 minutes, 54 seconds
    Time since path change: 34 seconds
    Number of LSP IDs (Tun_Instances) used: 55
  Current LSP: [ID: 55]
    Uptime: 34 seconds
  Prior LSP: [ID: 49]
    ID: path option unknown
    Removal Trigger: tunnel shutdown
Tun_Instance: 55
Segment-Routing Path Info (isis  level-1)
  Segment0[Link]: 192.168.2.1 - 192.168.2.2, Label: 46
  Segment1[Link]: 192.168.4.2 - 192.168.4.1, Label: 49
```

# Configuring Affinity on an Interface

Perform the following steps to configure affinity on an interface:

```
interface GigabitEthernet2
 ip address 192.168.2.1 255.255.255.0
 ip router isis 1
 negotiation auto
 mpls traffic-eng tunnels
 mpls traffic-eng attribute-flags 0x1
 isis network point-to-point
 ip rsvp bandwidth
```

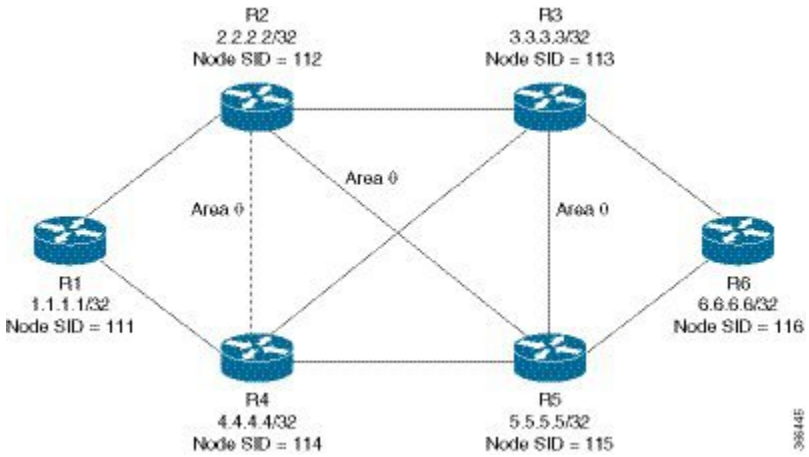
# Configuring Segment Routing Traffic Engineering With OSPF

Consider the following inter area and intra area use cases for configuring SR-TE with OSPF:

## Configuring Intra Area Tunnel

Consider the following topology to configure intra area tunnel:

*Figure 12: Intra Area Tunnel*




All the routers are configured in the same area, Area 0.

**Configuration at the head end router R1**:

```
router ospf 10
fast-reroute per-prefix enable prefix-priority low
fast-reroute per-prefix ti-lfa
segment-routing mpls
mpls traffic-eng router-id Loopback1
mpls traffic-eng area 0

interface GigabitEthernet2   //interface connecting to the router 2
ip address 100.101.1.1 255.255.255.0
ip ospf 10 area 0
ip ospf network point-to-point
negotiation auto
mpls traffic-eng tunnels

interface GigabitEthernet4  //interface connecting to the router 4
ip address 100.101.1.1 255.255.255.0
ip ospf 10 area 0
ip ospf network point-to-point
negotiation auto
mpls traffic-eng tunnels

interface loopback1
ip address 1.1.1.1/32
ip ospf 10 area 0
```

**Configuration at the tail-end router R6**:

```
router ospf 10
fast-reroute per-prefix enable prefix-priority low
fast-reroute per-prefix ti-lfa
segment-routing mpls
mpls traffic-eng area 0
mpls traffic-eng router-id Loopback1
interface GigabitEthernet2   //interface connecting to the router 3
ip address 100.101.2.1 255.255.255.0
ip ospf 10 area 0
ip ospf network point-to-point
negotiation auto
mpls traffic-eng tunnels
```

```
interface GigabitEthernet4  //interface connecting to the router 5
ip address 100.101.2.1 255.255.255.0
ip ospf 10 area 0
ip ospf network point-to-point
negotiation auto
mpls traffic-eng tunnels

interface loopback1
ip address 6.6.6.6/32
ip ospf 10 area 0
```

## Explicit Path SR-TE Tunnel 1

Consider tunnel 1 based only on IP addresses:

```
ip explicit-path name IP_PATH1
 next-address 2.2.2.2
 next-address 3.3.3.3
 next-address 6.6.6.6
!
interface Tunnel1
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng path-option 10 explicit name IP_PATH1 segment-routing
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10
end
```

## Explicit Path SR-TE Tunnel 2

Consider tunnel 2 based on node SIDs

```
ip explicit-path name IA_PATH
 next-label 114
 next-label 115
 next-label 116
!
interface Tunnel2
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng bandwidth 10000 class-type 1
 tunnel mpls traffic-eng path-option 10 explicit name NODE_PATH segment-routing
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10
end
```

## Explicit Path SR-TE Tunnel 3

Consider that tunnel 3 is based on a mix of IP addresses and label

```
ip explicit-path name MIXED_PATH enable
 next-address 2.2.2.2
 next-address 3.3.3.3
 next-label 115
 next-label 116
```

```
!
interface Tunnel3
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng path-option 10 explicit name MIXED_PATH segment-routing
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10
```

**Note** In the case of mixed path, IP next-hop cannot be used after using Node SIDs in the path. The following path will not be valid:

```
ip explicit-path name MIXED_PATH enable
next-label 115
next-label 116
next-address 2.2.2.2
```

### Dynamic Path SR-TE Tunnel 4

Consider that tunnel 4is based on adjacency SIDs

```
interface Tunnel4
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng bandwidth 10000 class-type 1
 tunnel mpls traffic-eng path-option 10 dynamic segment-routing
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10
end
```

### Dynamic Path SR-TE Tunnel 5

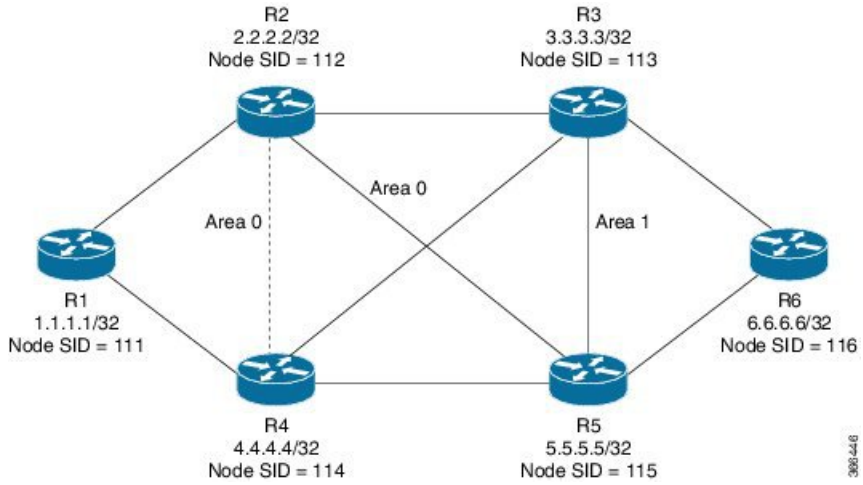
Consider that tunnel 5 is based on Node SIDs

```
 interface Tunnel5
 ip unnumbered Loopback1
 tunnel mode mpls traffic-eng
 tunnel destination 6.6.6.6
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 6 6
 tunnel mpls traffic-eng path-option 10 segment-routing
 tunnel mpls traffic-eng path-selection metric igp
 tunnel mpls traffic-eng load-share 10
```

## Configuring Inter Area Tunnel

Consider the following topology to configure inter area tunnel:

*Figure 13: Inter Area Tunnel*




All the routers are configured in the same area, area 0 except R6 which is configured in area 1.

**Configuration at the head end router R1**:

```
router ospf 10
fast-reroute per-prefix enable prefix-priority low
fast-reroute per-prefix ti-lfa
segment-routing mpls
mpls traffic-eng router-id Loopback1
mpls traffic-eng area 0

interface GigabitEthernet2   //interface connecting to the router 2
ip address 100.101.1.1 255.255.255.0
ip ospf 10 area 0
ip ospf network point-to-point
negotiation auto
mpls traffic-eng tunnels

interface GigabitEthernet4  //interface connecting to the router 4
ip address 100.101.1.1 255.255.255.0
ip ospf 10 area 0
ip ospf network point-to-point
negotiation auto
mpls traffic-eng tunnels

interface loopback1
ip address 1.1.1.1/32
ip ospf 10 area 0
```

**Configuration at the tail-end router R6**:

```
router ospf 10
fast-reroute per-prefix enable prefix-priority low
fast-reroute per-prefix ti-lfa
segment-routing mpls
mpls traffic-eng area 1
mpls traffic-eng router-id Loopback1
interface GigabitEthernet2   //interface connecting to the router 3
ip address 100.101.2.1 255.255.255.0
ip ospf 10 area 1
ip ospf network point-to-point
```

```
negotiation auto
mpls traffic-eng tunnels

interface GigabitEthernet4  //interface connecting to the router 5
ip address 100.101.2.1 255.255.255.0
ip ospf 10 area 1
ip ospf network point-to-point
negotiation auto
mpls traffic-eng tunnels

interface loopback1
ip address 6.6.6.6/32
ip ospf 10 area 1
```

### Restrictions for Configuring Inter Area Tunnel

The following are the restrictions for configuring inter area tunnel:

- The dynamic option with node and adjacency SID are not supported.
- You can configure inter are tunnel using the explicit path containing only labels and/or IP address and labels.

**Note** The IP address can be used only be till the Area Border Router (ABR) and after that you need to specify only the labels.

### Explicit Path SR-TE Tunnel 1

Consider tunnel 2 is based on node SIDs.

```
ip explicit-path name IA_PATH
next-label 114
next-label 115
next-label 116
!
interface Tunnel2
ip unnumbered Loopback1
tunnel mode mpls traffic-eng
tunnel destination 6.6.6.6
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng priority 6 6
tunnel mpls traffic-eng bandwidth 10000 class-type 1
tunnel mpls traffic-eng path-option 10 explicit name NODE_PATH segment-routing
tunnel mpls traffic-eng path-selection metric igp
tunnel mpls traffic-eng load-share 10
end
```

### Explicit Path SR-TE Tunnel 2

Consider that tunnel 3 is based on a mix of IP Addresses and label.

```
ip explicit-path name MIXED_PATH enable
next-address 2.2.2.2
next-address 3.3.3.3
next-label 115
next-label 116
!
```

```
interface Tunnel3
ip unnumbered Loopback1
tunnel mode mpls traffic-eng
tunnel destination 6.6.6.6
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng priority 6 6
tunnel mpls traffic-eng path-option 10 explicit name MIXED_PATH segment-routing
tunnel mpls traffic-eng path-selection metric igp
tunnel mpls traffic-eng load-share 10
```

# Verifying Configuration of the SR-TE Tunnels

Use the **show mpls traffic-eng tunnels** *tunnel-number* command to verify the configuration of the SR-TE tunnels.

# Verifying Tunnel 1

```
Name: R1_t1                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type explicit IP_PATH (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours
      Time since path change: 2 seconds
      Number of LSP IDs (Tun_Instances) used: 1814
    Current LSP: [ID: 1814]
      Uptime: 2 seconds
      Selection: reoptimization
    Prior LSP: [ID: 1813]
      ID: path option unknown
      Removal Trigger: configuration changed
  Tun_Instance: 1814
  Segment-Routing Path Info (ospf 10  area 0)
    Segment0[Node]: 4.4.4.4, Label: 114
    Segment1[Node]: 5.5.5.5, Label: 115
    Segment2[Node]: 6.6.6.6, Label: 116
```

•

# Verifying Tunnel 2

```
Name: R1_t2                             (Tunnel1) Destination: 6.6.6.6
```

```
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type explicit IA_PATH (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours, 1 minutes
      Time since path change: 1 seconds
      Number of LSP IDs (Tun_Instances) used: 1815
    Current LSP: [ID: 1815]
      Uptime: 1 seconds
    Prior LSP: [ID: 1814]
      ID: path option unknown
      Removal Trigger: configuration changed
  Tun_Instance: 1815
  Segment-Routing Path Info (ospf 10  area 0)
    Segment0[ - ]: Label: 114
    Segment1[ - ]: Label: 115
    Segment2[ - ]: Label: 116
```

# Verifying Tunnel 3

```
Name: R1_t3                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type explicit MIXED_PATH (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours, 2 minutes
      Time since path change: 2 seconds
      Number of LSP IDs (Tun_Instances) used: 1816
    Current LSP: [ID: 1816]
      Uptime: 2 seconds
      Selection: reoptimization
    Prior LSP: [ID: 1815]
      ID: path option unknown
      Removal Trigger: configuration changed
  Tun_Instance: 1816
  Segment-Routing Path Info (ospf 10  area 0)
```

```
  Segment0[Node]: 2.2.2.2, Label: 112
  Segment1[Node]: 3.3.3.3, Label: 113
  Segment2[ - ]: Label: 115
  Segment3[ - ]: Label: 116
```

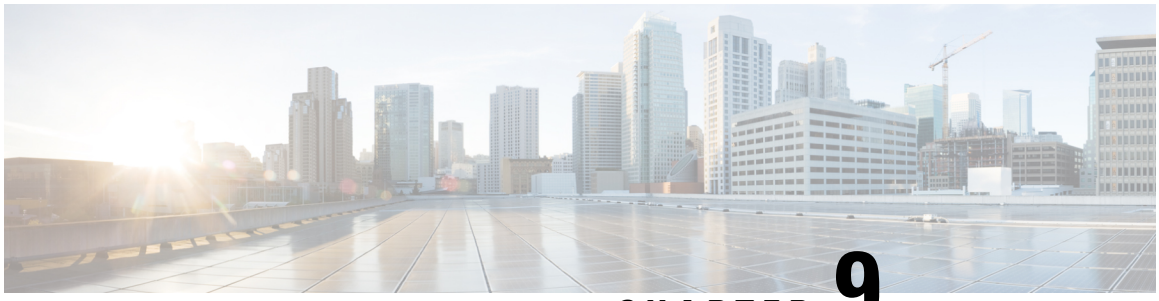
# Verifying Tunnel 4

```
Name: R1_t4                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) type dynamic (Basis for Setup, path weight 30)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: dynamic path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
      Time since created: 6 days, 19 hours
      Time since path change: 2 seconds
      Number of LSP IDs (Tun_Instances) used: 1813
    Current LSP: [ID: 1813]
      Uptime: 2 seconds
    Prior LSP: [ID: 1806]
      ID: path option unknown
      Removal Trigger: configuration changed
  Tun_Instance: 1813
  Segment-Routing Path Info (ospf 10  area 0)
    Segment0[Link]: 192.168.2.1 - 192.168.2.2, Label: 17
    Segment1[Link]: 192.168.4.2 - 192.168.4.1, Label: 25
    Segment2[Link]: 192.168.8.1 - 192.168.8.2, Label: 300
```

# Verifying Tunnel 5

```
Name: R1_t5                             (Tunnel1) Destination: 6.6.6.6
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, type segment-routing (Basis for Setup)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 6  6   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
     Protection: any (default)
    Path-invalidation timeout: 45000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 10 [200000000]
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: segment-routing path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  History:
    Tunnel:
```

```
    Time since created: 6 days, 19 hours, 4 minutes
    Time since path change: 14 seconds
    Number of LSP IDs (Tun_Instances) used: 1817
  Current LSP: [ID: 1817]
    Uptime: 14 seconds
    Selection: reoptimization
  Prior LSP: [ID: 1816]
    ID: path option unknown
    Removal Trigger: configuration changed
Tun_Instance: 1817
Segment-Routing Path Info (ospf 10  area 0)
  Segment0[Node]: 6.6.6.6, Label: 116
```



CHAPTER 9

# BGP Dynamic Segment Routing Traffic Engineering

Border Gateway Protocol (BGP) has become a popular choice as a routing protocol in Data Center (DC) network. The ability to setup Segment Routing-Traffic Engineering (SR-TE) path initiated by BGP simplifies DC network operation.


- Feature Information for BGP Dynamic Segment Routing Traffic Engineering, on page 109
- Restrictions for Segment Routing –Traffic-Engineering Dynamic BGP, on page 109
- Information About Segment Routing –Traffic-Engineering Dynamic BGP, on page 110
- How to Configure TE Label Switched Path Attribute-Set, on page 111


## Feature Information for BGP Dynamic Segment Routing Traffic Engineering

*Table 8: Feature Information for BGP Dynamic Segment Routing Traffic Engineering*

| Feature Name | Releases | Feature Information |
|---|---|---|
| BGP Dynamic Segment Routing Traffic Engineering | Cisco IOS XE Amsterdam 17.3.2 | In BGP dynamic SR-TE, the label Switched Path (LSP) is enabled on demand when defined criteria and policies are met.<br>The following commands were introduced or modified:<br>**mpls traffic-eng lsp attribute** *name* |

## Restrictions for Segment Routing –Traffic-Engineering Dynamic BGP

- For Anycast SID support to work BGP-TE should be configured with the prepend feature.
- In the case of BGP Dynamic SR-TE if SR-TE fails, forwarding gets broken.

# Information About Segment Routing –Traffic-Engineering Dynamic BGP

In BGP dynamic SR-TE, the label Switched Path (LSP) is enabled on demand when defined criteria and policies are met and that is the key difference between manually enabled SR-TE and BGP dynamic SR-TE. Policies, for example, low latency path, minimum cost path, and so on are carried via BGP and matches on a given customer prefix. SR-TE tunnel used for L3VPN or Virtual Private LAN Services (VPLS) using BGP for auto-discovery and signaling is referred to as BGP-TE Dynamic.

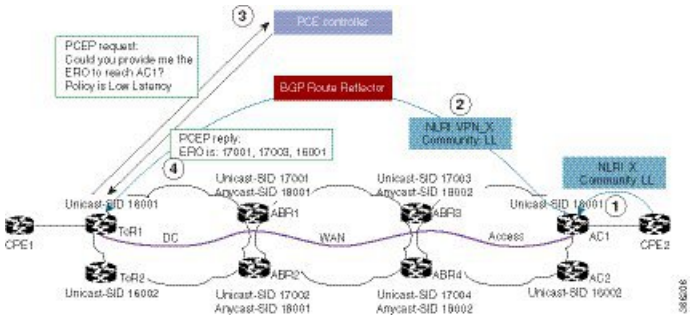
BGP SR-TE dynamic assumes the on-demand auto-tunnel resides in single IGP domain. In this case path computation is done via IGP. SR-TE auto-tunnel created based on the request from BGP is a dynamic SR-TE tunnel. In other words, tunnel path information, or label stack, is computed based on the BGP next-hop and TE attribute configuration. BGP dynamic SR-TE functions to trigger an On-demand LSP (auto-tunnel). The functions include:

- Tag customer prefixes (IPv4 or L3VPN VRF) using communities (community list) via route map configuration.
- Associate each community with a TE attribute-set or profile.

SR-TE profile is locally configured in attribute-set to define certain SR-TE parameters, for example, latency, disjoint path and so on. Once the BGP customer prefixes are mapped to an SR-TE-profile, a tunnel is dynamically created (auto-tunnel or On demand Label Switched Path (LSP)) using the parameters defined in the attribute-set, for each specified BGP next-hop and attribute-set pair associated with the prefixes. A binding SID is associated with each SR-TE auto-tunnel and passed to BGP. The binding SID or binding label is installed into Routing Information Base (RIB) and Forwarding Information Base (FIB). FIB resolves BGP path via the binding SID or binding label, which forwards over the On demand SR-TE auto-tunnel. The binding-SID is also used to steer the customer traffic over the SR-TE LSP.

It must be noted that BGP only carries the SR-TE policy in this case, while path computation is done via IGP in a single IGP domain. In a single IGP domain the headend node has full visibility of the end to end path and the topology engineering database (Traffic Engineering Database or TED). Also it is assumed with BGP Dynamic SR-TE that all the nodes reside within single AS and single IGP domain.

***Figure 14: BGP-TE Dynamic Workflow***




The above figure depicts the workflow for BGP-TE dynamic using multiple routing domains use case:

1. Customer premise equipment 2 (CPE) sends BGP update for Prefix-X and adds LL community, for example, 100:333.
2. AC1 announces a VPN route for prefix X with LL community.

3. After receiving BGP update of the VPN route matching community LL, ToR1 sends a request to PCE controller for LSP path towards AC1 with low latency TE policy.
4. Path calculation element (PCE) controller replies with a label stack, for example, 17003, 1600.
5. ToR1 creates SR-TE auto-tunnel and installs the route for Prefix-X in VRF of this VPN.

## TE Label Switched Path Attribute-Set

TE-LSP attribute-set is used to configure the properties of a LSP. It describes TE profile or policy such as bandwidth, affinities inclusion and exclusion, links/nodes/SRLG inclusion and exclusion, metrics, path disjoint degree and group, and so on that are used to create an auto-tunnel.

# How to Configure TE Label Switched Path Attribute-Set

## Configuring TE Label Switched Path Attribute-Set

You can use the command **mpls traffic-eng lsp attribute** <*name*> to configure TE-LSP attribute. The following options are available:

```
Mpls traffic-eng lsp attribute name
   affinity      Specify attribute flags for links comprising LSP
   lockdown      Lockdown the LSP--disable reoptimization
   priority      Specify LSP priority
```

TE-LSP attribute command can be extended to support configuration for the two options **pce** and **path-selection**. It can be configured as following:

```
mpls traffic-eng lsp attribute name <test>
    path-selection
       metric <te/igp>
       invalidation <time-out> <drop/tear>
       segment-routing adjacency <protected/unprotected>
```

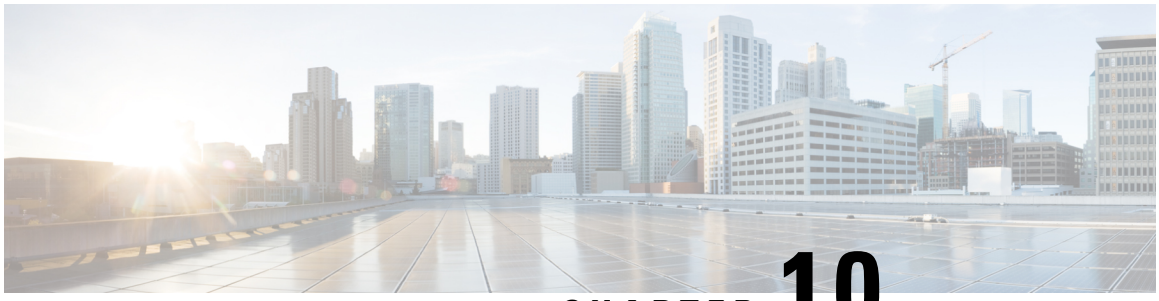
- If pce option is set in the TE attribute the dynamic path is calculated by PCE. Otherwise, the path is calculated locally by TE PCALC (path-calculation) entity. In the later case, IGP has to be configured and the BGP next-hop has to be both advertised by IGP and reachable from the local node over an IGP route.
- The option path-selection metric indicates whether the path calculation is based on TE metrics or IGP metrics. If this option is not configured the global value configured under mpls traffic-eng path-selection metric is used.
- The option **path-selection invalidation** configures the behavior of how an LSP reacts to soft failure from network. When an LSP path has a protected path from IGP against a link or node failure, the failure to the link or node is considered as soft failure.
- The option **path-selection segment-routing adjacency** indicates whether to choose an adjacency-SID with or without IGP protection when calculating LSP label stack.

- The option **pce disjoint-path** indicates the tunnel LSP is a member of disjoint-path group. Any LSPs within the same disjoint-path group do not traverse the same resources, such as links, nodes, or SRLG, in its path. This is used to create two or more tunnel LSPs with disjoint paths.

For BGP-TE Dynamic, a TE attribute name is associated with a BGP route-map set extension as following:

```
route-map <name>
    match community <name>
         set attribute-set  <name>
```

BGP uses the **attribute-set** <*name*> string together with its BGP next-hop to request a SR-TE auto-tunnel.



CHAPTER 10

# Segment Routing On Demand Next Hop for L3/L3VPN

When redistributing routing information across domains, provisioning of multi-domain services (L2VPN & L3VPN) has its own complexity and scalability issues. On Demand Next Hop (ODN) triggers delegation of computation of an end-to-end LSP to a PCE controller including constraints and policies without doing any redistribution. It then installs the replied multi-domain LSP for the duration of the service into the local forwarding information base (FIB).


- Feature Information for Segment Routing On Demand Next Hop for L3/L3VPN, on page 113
- Restrictions for Segment Routing On Demand SR PFP ODN AUTO STEERING (PCE DELEGATED) for L3/L3VPN, on page 114
- Information About Segment Routing On Demand SR PFP ODN AUTO STEERING (PCE DELEGATED) for L3/L3VPN, on page 114
- SR-TE Policy, Color Extended Community, Affinity Constraint, and Disjointness Constraint, on page 115
- How to Configure Segment Routing On Demand Next Hop for L3/L3VPN, on page 117
- Verifying Segment Routing On Demand Next Hop for L3/L3VPN, on page 120
- Configuring Color Extended Community, Affinity Constraint, and Disjointness Constraint, on page 124
- Verifying SR-TE ODN Color Extended Community, Affinity Constraint, and Disjointness Constraint, on page 126
- Troubleshooting the SR-TE ODN Color Extended Community, Affinity Constraint, and Disjointness Constraint, on page 131


# Feature Information for Segment Routing On Demand Next Hop for L3/L3VPN

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

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

*Table 9: Feature Information for Segment Routing On Demand Next Hop for L3/L3VPN*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing On Demand Next Hop for L3/L3VPN | Cisco IOS XE Amsterdam 17.3.2 | On-Demand Next Hop (ODN) triggers delegation of computation of an end-to-end LSP to a PCE controller including constraints and policies without doing any redistribution.<br>The following commands were introduced or modified:<br>**route-map BGP_TE_MAP permit**, **mpls traffic-eng tunnels**, **sh bgp li li summary**, **sh pce client peer**, **sh pce ipv4 peer**, **sh ip route vrf sr**, **sh ip bgp vpnv4 vrf sr**, **sh ip cef label-table**, **sh mpls traffic-eng tunnels**, **sh pce client lsp brief**, **sh pce lsp summ**, **sh pce lsp det**, **routing-default-optimize** |
| SR-TE Policy, Color Extended Community, Affinity Constraint, and Disjointness Constraint | Cisco IOS XE Amsterdam 17.3.2 | A new command **segment-routing traffic-eng** is added to configure the SR policy under segment routing. Also, the configuration of affinity and disjointness constraints is supported.<br>Support for ODN with color extended community is introduced. |

# Restrictions for Segment Routing On Demand SR PFP ODN AUTO STEERING (PCE DELEGATED) for L3/L3VPN

- On Demand Next Hop (ODN) anycast SID is not supported.
- ODN for IPv6 is not supported.
- SR ODN tunnel is not supported with BGP Nonstop Routing (NSR). It is only supported with BGP Nonstop Forwarding (NSF).

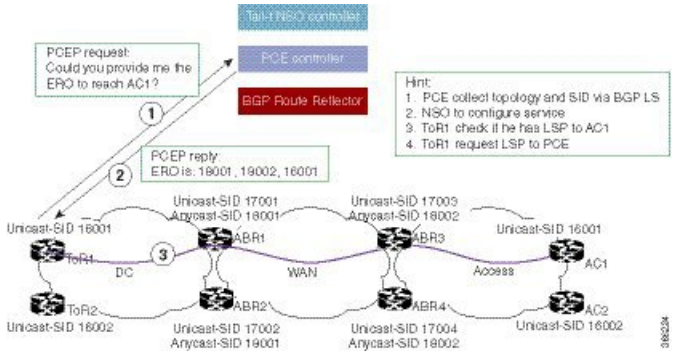
  To enable BGP NSF, use the following command:

```
bgp grace-full restart
neighbor 10.0.0.2 ha-mode graceful-restart
```

# Information About Segment Routing On Demand SR PFP ODN AUTO STEERING (PCE DELEGATED) for L3/L3VPN

On Demand SR PFP ODN AUTO STEERING (PCE DELEGATED) leverages upon BGP Dynamic SR-TE capabilities and adds the path computation (PCE) ability to find and download the end to end path based on the requirements. ODN triggers an SR-TE auto-tunnel based on the defined BGP policy. As shown in the below figure, an end to end path between ToR1 and AC1 can be established from both ends based on low latency or other criteria for VRF (L3VPN) or IPv4 services. The work-flow for ODN is summarized as follows:

*Figure 15: ODN Operation*




1. PCE controller collects topology and SIDs information via BGP Link State (BGP-LS). For more information on BGP-LS, refer BGP Link-State.
2. If NSO controller is enable, it configures L3VPN VRF or IPv4 prefixes and requests are sent to ToR1 and AC1.
3. ToR1 and AC1 checks if a LSP towards each other exists. If not, a request is sent to the PCE controller to compute that SR-TE path that matches SR-TE policy that is carried via BGP.
4. PCE controller computes the path and replies with a label stack (18001, 18002, 16001, example in ToR1).
5. ToR1 and AC1 create a SR-TE auto-tunnel and reply back to the NSO controller indicating that the LSP for VRF or IPv4 is up and operational.

# SR-TE Policy, Color Extended Community, Affinity Constraint, and Disjointness Constraint

- MPLS TE new SR-TE policy command—**segment-routing traffic-eng**
- Color-extended community
- Affinity constraints
- Disjointness constraints

## SR-TE Policy Command

## Color Extended Community

In earlier releases, the router created segment routed Traffic Engineering (SR-TE) tunnels based on a tunnel-profile or attribute set. As part of this functionality, an inbound route-map with a “match community” and “set attribute-set” was added on the ingress node and the route-map matched against communities received in the BGP updates. A BGP update with a matching community would initiate an SR-TE tunnel for the nexthop TE-profile.

- An SR-TE policy is created on the ingress router for the Color-Endpoint pair.

- The egress router adds the ‘color extended’ community to the BGP updates that require a Traffic-Engineered path.

# Affinity Constraint

Affinity is a 32-bit constraint used by the PCE and PCALC for calculating paths that take the "affinity constraint" into account.

Affinity constraints let you assign, or map, color names for path affinities. After mappings are defined, the attributes can be referred to by the corresponding color name in the command.

Affinity maps are used to map operator-defined color names to a bit position in the affinity bitmap.

Supported Affinity constraints are:

- include-all—indicates that constrained shortest path first (CSPF) includes a link when calculating a path, only if each link administrative group bit has the same name as each affinity bit.
- include-any—indicates that CSPF includes a link when calculating a path, if at least one link administrative group bit has the same name as an affinity bit.
- exclude-any—indicates that CSPF excludes a link when calculating a path, if any link administrative group bit has the same name as an affinity bit.

# Disjointness Constraint

Disjointness is used to describe two or more services that must be completely disjoint of each other. Disjointness is useful for providing traffic flow redundancy in the network.

Disjointness is controlled by the PCE. The PCE learns of the network topology through an IGP (OSPF or IS-IS) through the BGP-LS protocol and is capable of computing paths based on the IGP or TE metric.

The PCE uses the disjoint policy to compute two lists of segments that steer traffic from the source node towards the destination node along disjoint paths. Disjoint paths can originate from either the same or different head-ends.

A "disjoint level" refers to the type of resources that should not be shared by the two computed paths. The PCE supports the following disjoint path computations:

- Link
- Node
- Shared risk link group (SRLG)

When the first request is received from Path Computation client (PCC) or an ingress node, with a given disjoint-group ID, a list of segments is computed based on the metric requested, encoding the shortest path from source to destination.

When the second request is received with the same disjoint-group ID, based on the information received in both requests, the PCE computes two disjoint paths from the source to the destination.

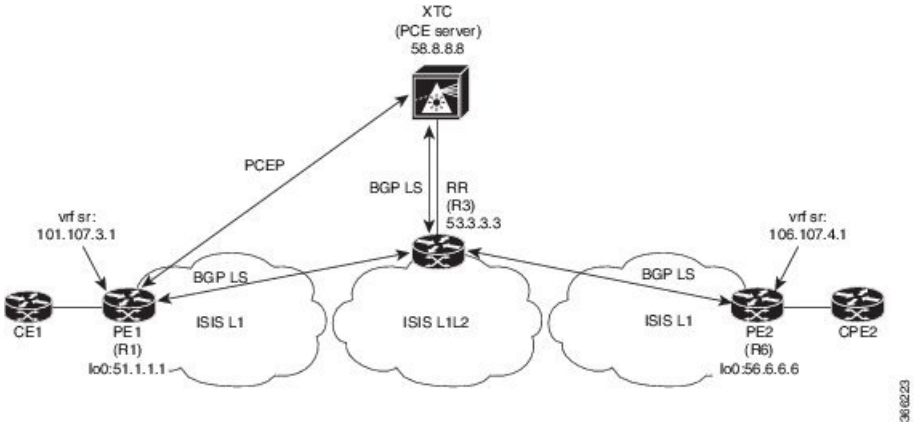
Both paths are computed at the same time. The shortest list of segments is calculated to steer traffic on the computed paths.

# How to Configure Segment Routing On Demand Next Hop for L3/L3VPN

## Configuring Segment Routing On Demand Next Hop for L3/L3VPN

Perform the following steps to configure on-demand next hop for SR-TE. The below figure is used as a reference to explain the configuration steps.

***Figure 16: ODN Auto-Tunnel Setup***




1. Configure the router (R6 tail end) with VRF interface.

```
interface GigabitEthernet0/2/2
 vrf forwarding sr
 ip address 10.0.0.1 255.0.0.0
 negotiation auto

interface Loopback0
 ip address 192.168.0.1 255.255.0.0
 ip router isis 1
```

2. Tags VRF prefix with BGP community on R6 (tail end).

```
route-map BGP_TE_MAP permit 9
 match ip address traffic
 set community 3276850

ip access-list extended traffic
 permit ip 10.0.0.1 255.255.0.0 any
```

3. Enable BGP on R6 (tail end) and R1 (head end) to advertise and receive VRF SR prefix and match on community set on R6 (tail end).

```
 router bgp 100
 bgp router-id 172.16.0.1
 bgp log-neighbor-changes
 bgp graceful-restart
 no bgp default ipv4-unicast
 neighbor 10.0.0.2 remote-as 100
```

```
 neighbor 10.0.0.2 update-source Loopback0

 address-family ipv4
  neighbor 10.0.0.2 activate
  neighbor 10.0.0.2 send-community both
  neighbor 10.0.0.2 next-hop-self
 exit-address-family

 address-family vpnv4
  neighbor 10.0.0.2 activate
  neighbor 10.0.0.2 send-community both
  neighbor 10.0.0.2 route-map BGP_TE_MAP out
 exit-address-family

 address-family link-state link-state
  neighbor 10.0.0.2 activate
 exit-address-family

 address-family ipv4 vrf sr
  redistribute connected
 exit-address-family

route-map BGP_TE_MAP permit 9
 match ip address traffic
 set community 3276850

ip access-list extended traffic
 permit ip 10.0.0.1 255.255.0.0 any


router bgp 100
 bgp router-id 192.168.0.2
 bgp log-neighbor-changes
 bgp graceful-restart
 no bgp default ipv4-unicast
 neighbor 10.0.0.2 remote-as 100
 neighbor 10.0.0.2 update-source Loopback0


address-family ipv4
  neighbor 10.0.0.2 activate
  neighbor 10.0.0.2 send-community both
  neighbor 10.0.0.2 next-hop-self
 exit-address-family

 address-family vpnv4
  neighbor 10.0.0.2 activate
  neighbor 10.0.0.2 send-community both
  neighbor 10.0.0.2 route-map BGP_TE_MAP in
 exit-address-family

 address-family link-state link-state
  neighbor 10.0.0.2 activate
 exit-address-family

 address-family ipv4 vrf sr
  redistribute connected
 exit-address-family

route-map BGP_TE_MAP permit 9
 match community 1
 set attribute-set BGP_TE5555

ip community-list 1 permit 3276850
```

```
mpls traffic-eng lsp attributes BGP_TE5555
 path-selection metric igp
 pce
```

4. Enable PCE and auto-tunnel configurations on R1.

```
mpls traffic-eng tunnels
mpls traffic-eng pcc peer 10.0.0.3 source 10.0.0.4 precedence 255
mpls traffic-eng auto-tunnel p2p tunnel-num min 2000 max 5000
```

5. Enable all core links with SR-TE configurations and ensure that they are enabled as point to point interfaces.

```
mpls traffic-eng tunnels

interface GigabitEthernet0/2/0
 ip address 101.102.6.1 255.255.255.0
 ip router isis 1
 mpls traffic-eng tunnels
 isis network point-to-point

interface GigabitEthernet0/3/1
 vrf forwarding sr
 ip address 101.107.3.1 255.255.255.0
 negotiation auto

end
```

6. Enable R3 (RR) to advertise TED to the PCE server via BGP-LS.

```
router isis 1
 net 49.0002.0000.0000.0003.00
 ispf level-1-2
 metric-style wide
 nsf cisco
 nsf interval 0
 distribute link-state
 segment-routing mpls
 segment-routing prefix-sid-map advertise-local
 redistribute static ip level-1-2
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng level-1
 mpls traffic-eng level-2

router bgp 100
 bgp router-id 10.0.0.2
 bgp log-neighbor-changes
 bgp graceful-restart
 no bgp default ipv4-unicast
 neighbor 10.0.0.3 remote-as 100
 neighbor 10.0.0.3 update-source Loopback0

 address-family ipv4
 neighbor 10.0.0.3 activate
 exit-address-family
```

7. Enable PCE server configuration and verify BGP-LS session is properly established with RR.

```
Device# sh bgp li li summary
BGP router identifier 10.0.0.3, local AS number 100
BGP generic scan interval 60 secs
Non-stop routing is enabled
BGP table state: Active
Table ID: 0x0   RD version: 1436
BGP main routing table version 1436
```

```
BGP NSR Initial initsync version 1 (Reached)
BGP NSR/ISSU Sync-Group versions 0/0
BGP scan interval 60 secs
BGP is operating in STANDALONE mode.
Process       RcvTblVer   bRIB/RIB   LabelVer  ImportVer  SendTblVer  StandbyVer
Speaker            1436       1436          1436       1436           1436
       0

Neighbor        Spk    AS MsgRcvd   MsgSent   TblVer  InQ  OutQ  Up/Down  St/PfxRcd
10.0.0.2          0      100   19923       17437        1436       0     0
1w2d        103

Device# sh pce ipv4 topo | b Node 3
Node 3
  TE router ID: 10.0.0.2
  Host name: R3
  ISIS system ID: 0000.0000.0003 level-1

  ISIS system ID: 0000.0000.0003 level-2
  Prefix SID:
    Prefix 10.0.0.2, label 20011 (regular)
```

# Verifying Segment Routing On Demand Next Hop for L3/L3VPN

The ODN verifications are based on L3VPN VRF prefixes.

1. Verify that PCEP session between R1 (headend and PCE server) is established.

```
Device# sh pce client peer
PCC's peer database:
--------------------
Peer address: 10.0.0.3 (best PCE)
  State up
  Capabilities: Stateful, Update, Segment-Routing
```

2. Verify that PCEP session is established between all the peers (PCCs).

```
Device# sh pce ipv4 peer
PCE's peer database:
--------------------
Peer address: 10.0.0.4
  State: Up
  Capabilities: Stateful, Segment-Routing, Update
Peer address: 172.16.0.5
  State: Up
  Capabilities: Stateful, Segment-Routing, Update
```

3. Verify that R1 (headend) has no visibility to R6 loopback address.

```
Device# sh ip route 192.168.0.1
% Network not in table
```

4. Verify that VRF prefix is injected via MP-BGP in R1 VRF SR routing table.

```
Device# sh ip route vrf sr
Routing Table: sr
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
```

```
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR
Gateway of last resort is not set
     10.0.0.6/8 is variably subnetted, 2 subnets, 2 masks
C        10.0.0.7/24 is directly connected, GigabitEthernet0/3/1
L        10.0.0.7/32 is directly connected, GigabitEthernet0/3/1
      10.0.0.8/24 is subnetted, 1 subnets
B        10.0.0.9 [200/0] via binding label: 865, 4d21h
```

5. Verify that BGP is associating properly the policy and binding SID with the VRF prefix.

```
Device# sh ip bgp vpnv4 vrf sr 106.107.4.0
BGP routing table entry for 100:100:106.107.4.0/24, version 3011
Paths: (1 available, best #1, table sr)
  Not advertised to any peer
  Refresh Epoch 4
  Local
    192.168.0.1 (metric 10) (via default) from 10.0.0.2 (10.0.0.2)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Community: 3276850
      Extended Community: RT:100:100
      Originator: 192.168.0.1, Cluster list: 10.0.0.2
      mpls labels in/out nolabel/1085
      binding SID: 865 (BGP_TE5555)
      rx pathid: 0, tx pathid: 0x0
```

6. Verify binding label association with VRF prefix.

```
Device# sh ip route vrf sr 106.107.4.0
Routing Table: sr
Routing entry for 106.107.4.0/24
  Known via "bgp 100", distance 200, metric 0, type internal
  Routing Descriptor Blocks:
  *  Binding Label: 865, from 10.0.0.2, 4d22h ago
      Route metric is 0, traffic share count is 1
      AS Hops 0
      MPLS label: 1085
      MPLS Flags: NSF
```

7. Verify that VRF prefix is forwarded via ODN auto-tunnel.

```
Device# sh ip cef label-table
Label               Next Hop             Interface
0                    no route
865                  attached             Tunnel2000

     Device# sh ip cef vrf sr 106.107.4.0 detail
10.0.0.8/24, epoch 15, flags [rib defined all labels]
  recursive via 865 label 1085
    attached to Tunnel2000
```

8. Verify ODN auto-tunnel status.

```
Device# sh mpls traffic-eng tunnels
P2P TUNNELS/LSPs:
Name: R1_t2000                 (Tunnel2000) Destination: 192.168.0.1 Ifhandle: 0x6F5
(auto-tunnel for BGP TE)
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected---
auto-tunnel 2000
    path option 1, (SEGMENT-ROUTING) (PCE) type dynamic (Basis for Setup, path weight
 10)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF
    Metric Type: IGP (interface)
    Path Selection:
```

```
     Protection: any (default)
    Path-selection Tiebreaker:
      Global: not set   Tunnel Specific: not set   Effective: min-fill (default)
    Hop Limit: disabled
    Cost Limit: disabled
    Path-invalidation timeout: 10000 msec (default), Action: Tear
    AutoRoute: disabled LockDown: disabled Loadshare: 0 [0] bw-based
    auto-bw: disabled
    Attribute-set: BGP_TE5555---□ attribute-set
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: dynamic path option 1 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
  PCEP Info:
    Delegation state: Working: yes   Protect: no
    Working Path Info:
      Request status: processed
      Created via PCRep message from PCE server: 10.0.0.3--□ via PCE server
      PCE metric: 30, type: IGP
    Reported paths:
      Tunnel Name: Tunnel2000_w
       LSPs:
        LSP[0]:
         source 10.0.0.4, destination 192.168.0.1, tunnel ID 2000, LSP ID 1
         State: Admin up, Operation active
         Binding SID: 865
         Setup type: SR
         Bandwidth: requested 0, used 0
         LSP object:
           PLSP-ID 0x807D0, flags: D:0 S:0 R:0 A:1 O:2
         Metric type: IGP, Accumulated Metric 0
         ERO:
           SID[0]: Adj, Label 2377, NAI: local 101.102.6.1 remote 10.0.0.10
           SID[1]: Unspecified, Label 17, NAI: n/a
           SID[2]: Unspecified, Label 20, NAI: n/a
  History:
    Tunnel:
      Time since created: 4 days, 22 hours, 21 minutes
      Time since path change: 4 days, 22 hours, 21 minutes
      Number of LSP IDs (Tun_Instances) used: 1
    Current LSP: [ID: 1]
      Uptime: 4 days, 22 hours, 21 minutes
  Tun_Instance: 1
  Segment-Routing Path Info (isis  level-1)
    Segment0[Link]: 101.102.6.1 - 10.0.0.10, Label: 2377
    Segment1[ - ]: Label: 17
    Segment2[ - ]: Label: 20
```

**9.** Verify ODN auto-tunnel LSP status on R1 (headend).

```
Device# sh pce client lsp brief
PCC's tunnel database:
----------------------
 Tunnel Name: Tunnel2000_w
   LSP ID 1
 Tunnel Name: Tunnel2000_p

R1# sh pce client lsp detail
PCC's tunnel database:
----------------------
Tunnel Name: Tunnel2000_w
 LSPs:
  LSP[0]:
   source 10.0.0.4, destination 192.168.0.1, tunnel ID 2000, LSP ID 1
   State: Admin up, Operation active
```

```
  Binding SID: 865
  Setup type: SR
  Bandwidth: requested 0, used 0
  LSP object:
    PLSP-ID 0x807D0, flags: D:0 S:0 R:0 A:1 O:2
  Metric type: IGP, Accumulated Metric 0
  ERO:
    SID[0]: Adj, Label 2377, NAI: local 101.102.6.1 remote 10.0.0.10
    SID[1]: Unspecified, Label 17, NAI: n/a
    SID[2]: Unspecified, Label 20, NAI: n/a
```

**10.** Verify ODN LSP status on the PCE server.

```
Device# sh pce lsp summ

PCE's LSP database summary:
--------------------------------
All peers:
 Number of LSPs:         1
  Operational: Up:       1 Down:                 0
  Admin state: Up:       1 Down:                 0
  Setup type: RSVP:      0 Segment routing:      1


Peer 10.0.0.4:
 Number of LSPs:         1
  Operational: Up:       1 Down:                 0
  Admin state: Up:       1  Down:                 0
  Setup type: RSVP:       0 Segment routing:      1
```

**11.** Verify detailed LSP information on the PCE server.

```
Device# sh pce lsp det
PCE's tunnel database:
----------------------
PCC 10.0.0.4:
Tunnel Name: Tunnel2000_w
 LSPs:
  LSP[0]:
   source 10.0.0.4, destination 192.168.0.1, tunnel ID 2000, LSP ID 48
   State: Admin up, Operation active
   Binding SID: 872
   PCEP information:
     plsp-id 526288, flags: D:1 S:0 R:0 A:1 O:2
   Reported path:
     Metric type: IGP, Accumulated Metric 0
      SID[0]: Adj, Label 885, Address: local 10.0.0.9 remote 10.0.0.10
      SID[1]: Unknown, Label 17,
      SID[2]: Unknown, Label 20,
   Computed path:
     Computed Time: Tue Dec 20 13:12:57 2016 (00:11:53 ago)
     Metric type: IGP, Accumulated Metric 30
      SID[0]: Adj, Label 885, Address: local 10.0.0.9 remote 10.0.0.10
      SID[1]: Adj, Label 17, Address: local 10.0.0.12 remote 10.0.0.13
      SID[2]: Adj, Label 20, Address: local 10.0.0.14 remote 10.0.0.14
   Recorded path:
     None
```

**12.** Shutdown the interface that is connected to VRF SR so that the prefix is no longer advertised by MP-BGP.

```
Device# int gig0/2/2
Device(config-if)#shut
```

**13.** Verify that VRF prefix is no longer advertised to R1 (headend) via R6 (tailend).

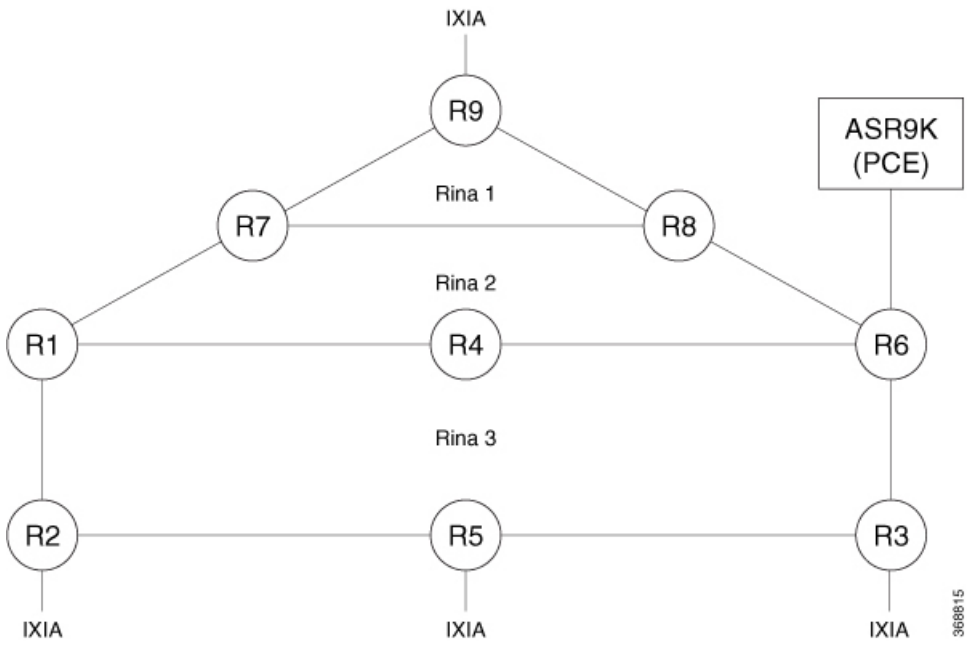
```
Device# sh ip route vrf sr
Routing Table: sr
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR
Gateway of last resort is not set
      10.0.0.6/8 is variably subnetted, 2 subnets, 2 masks
C        10.0.0.7/24 is directly connected, GigabitEthernet0/3/1
L        10.0.0.8/32 is directly connected, GigabitEthernet0/3/1
```

**14.** Verify that no ODN auto-tunnel exists.

```
Device# sh mpls traffic-eng tunnels
P2P TUNNELS/LSPs:
P2MP TUNNELS:
P2MP SUB-LSPS:
```

# Configuring Color Extended Community, Affinity Constraint, and Disjointness Constraint

Consider the following topology:




## Configuring Color Extended Community

**SR-TE Policy Headend Configuration on Node R3**

```
segment-routing traffic-eng
 on-demand color 100
  authorize restrict
   ipv4 prefix-list R9350_BGP_INTER_DOMAIN
  candidate-paths
   preference 1
    constraints
     segments
      dataplane mpls
      !
    !
dynamic
     pcep
    !
   !
  !
 !
pcc
  pce address <pce loopback ip>source-address <pcc loopback ip>
 !
```

### SR-TE Policy Taileend Configuration on Node R9

```
route-map R9_R3_R5_R2_BGP_INTER_DOMAIN permit 10
 match ip address prefix-list R9350_BGP_INTER_DOMAIN
 set extcommunity color 100   ---------------  Extended Color community configuration
route-map R9_R3_R5_R2_BGP_INTER_DOMAIN permit 20
ip prefix-list R9350_BGP_INTER_DOMAIN seq 35 permit 50.0.0.0/11 le 32
router bgp 1
 address-family vpnv4
  neighbor 201.201.201.201 activate
  neighbor 201.201.201.201 send-community both
  neighbor 201.201.201.201 route-map R9_R3_R5_R2_BGP_INTER_DOMAIN out
  neighbor 206.206.206.206 activate
  neighbor 206.206.206.206 send-community both
  neighbor 206.206.206.206 route-map R9_R3_R5_R2_BGP_INTER_DOMAIN out
 exit-address-family
 !
```

In the SR-TE ODN color template, to select the metric type, choose either **igp** or **te**:

```
Router(config-srte-odn-path-pref-dyn-metric)# type ?
  igp  Specify IGP metric
  te   Specify TE metric
```

# Configuring Affinity Constraint

```
segment-routing traffic-eng
 interface GigabitEthernet0/2/3
  affinity
   name 1
 on-demand color 100
  authorize restrict
   ipv4 prefix-list R9350_BGP_INTER_DOMAIN
  candidate-paths
   preference 1
    constraints
     segments
      dataplane mpls
     !
 affinity  ---------------------  Affinity configuration
      include-any  ------------  Affinity Type configuration
       name 1  ----------------  Affinity Name configuration
       !
```

```
    !
   dynamic
    pcep
   !
  !
 !
!
pcc
 pce address <pce loopback ip> source-address <pcc loopback ip>
!
affinity-map   ----------------------  Affinity Map configuration
  name 1 bit-position 1
```

## Configuring Disjointness Constraint

```
segment-routing traffic-eng
 on-demand color 100
  authorize restrict
   ipv4 prefix-list R9350_BGP_INTER_DOMAIN
  candidate-paths
   preference 1
    constraints
     segments
      dataplane mpls
     !
     affinity
      include-any
       name 1
         !
     !
     association-group   ---------------------  Disjointness configuration
       identifier 1
disjointness type node   ---------------------  Disjointness Type configuration
     source 1.0.0.0
      !
    !
    dynamic
     pcep
    !
   !
  !
 !
 pcc
  pce address <pce loopback ip> source-address <pcc loopback ip>
 !
 affinity-map
   name 1 bit-position 1
```

# Verifying SR-TE ODN Color Extended Community, Affinity Constraint, and Disjointness Constraint

**Note** C8xxx=C8200/C8300/C8500 or C8000v

**SR-TE Policy Name: 209.209.209.209|100**

```
C8xxx# show segment-routing traffic-eng policy name 209.209.209.209|100
Name: 209.209.209.209|100 (Color: 100 End-point: 209.209.209.209)
  Status:
    Admin: up, Operational: up for 51:34:38 (since 01-07 06:19:08.040)-□ Policy state is
UP
  Candidate-paths:
    Preference 1:
      Constraints:
        Affinity:
          include-any:    -------------□ Affinity Type
           1              -------------□ Affinity Name
        Disjointness information:
         Group ID: 1, Source: 1.0.0.0
         Type: Node Disjointness  ------□ Disjointness Type
        Dynamic (pce 12.12.12.12) (active) ---------□ PCE Computed Candidate-path
        Weight: 0, Metric Type: TE -----------------□ Metric Type
        Metric Type: TE, Path Accumulated Metric: 53  -□ Total IGP Metric from Source to
Destination
         18010 [Prefix-SID, 202.202.202.202]   ------------------|
         18007 [Prefix-SID, 211.211.211.211]   ------------------|                    |
           18002 [Prefix-SID, 207.207.207.207] -----------------□ This Segment List should
 follow Affinity path
         21 [Adjacency-SID, 10.10.20.2 - 10.10.20.1] ----------------|

Attributes:
    Binding SID: 87 ----------------□ Binding SID Allocated
      Allocation mode: dynamic
      State: Programmed
  Auto-policy info:
    Creator: BGP SR Policy Client
    IPv6 caps enable: yes
```

### To view detailed information about SR-TE Policy 209.209.209.209|100

```
C8xxx# show segment-routing traffic-eng policy name 209.209.209.209|100 detail
Name: 209.209.209.209|100 (Color: 100 End-point: 209.209.209.209)
  Status:
    Admin: up, Operational: up for 00:04:19 (since 01-10 06:20:57.810)
  Candidate-paths:
    Preference 1:
      Constraints:
        Affinity:
          include-any:
           1

Disjointness information:
         Group ID: 1, Source: 1.0.0.0
         Type: Node Disjointness
      Dynamic (pce 12.12.12.12) (active)
        Weight: 0, Metric Type: TE
        Metric Type: TE, Path Accumulated Metric: 53
          18010 [Prefix-SID, 202.202.202.202]
          18007 [Prefix-SID, 211.211.211.211]
          18002 [Prefix-SID, 207.207.207.207]
          21 [Adjacency-SID, 10.10.20.2 - 10.10.20.1]
  Attributes:
    Binding SID: 87
      Allocation mode: dynamic
      State: Programmed
  Auto-policy info:
    Creator: BGP SR Policy Client
    IPv6 caps enable: yes
  Forwarding-ID: 65711 (0x44)   -------□ This FWD-ID is used for forwarding traffic
  Stats:
```

```
    Packets: 8893       Bytes: 852848  -------  This counter indicates traffic flowing
through this SRTE policy

Event history:   ---  This indicates event happened with this SRTE Policy

    Timestamp                   Client                  Event type              Context:
Value
    ---------                   ------                  ----------              -------:
-----
    01-06 05:59:26.096          BGP SR Policy Cl        Policy created          Name:
209.209.209.209|100
    01-06 05:59:26.096          BGP SR Policy Cl        Set colour              Colour: 100

    01-06 05:59:26.096          BGP SR Policy Cl        Set end point           End-point:
 209.209.209.209
    01-06 05:59:26.096          BGP SR Policy Cl        Set dynamic pce         Path option:
 dynamic pce
    01-06 05:59:26.480          FH Resolution           Policy state UP         Status:
PATH RESOLVED
    01-06 05:59:40.424          FH Resolution           REOPT triggered         Status:
REOPTIMIZED
    01-06 05:59:49.249          FH Resolution           REOPT triggered         Status:
REOPTIMIZED
    01-06 05:59:56.469          FH Resolution           REOPT triggered         Status:
REOPTIMIZED
    01-07 05:15:19.918          FH Resolution           Policy state DOWN       Status:
PATH NOT RESOLVED
    01-07 06:15:55.739          FH Resolution           Policy state UP         Status:
PATH RESOLVED
    01-07 06:16:08.552          FH Resolution           REOPT triggered         Status:
REOPTIMIZED
    01-07 06:19:08.040          FH Resolution           Policy state DOWN       Status:
PATH NOT RESOLVED
    01-10 06:20:57.810          FH Resolution           Policy state UP         Status:
PATH RESOLVED
    01-10 06:21:05.211          FH Resolution           REOPT triggered         Status:
REOPTIMIZED
    01-10 06:21:08.036          FH Resolution           REOPT triggered         Status:
REOPTIMIZED
    01-10 06:21:10.073          FH Resolution           REOPT triggered         Status:
REOPTIMIZED
```

To check if the Affinity constraint is working, shut down any of the interfaces falling under the Affinity-defined path. If the constraint works, the SR-TE policy goes down instead of taking the another path (if available) to reach to the destination.

To check if the disjointness constraint is working, check the SR-TE policy information given by the PCE, which consists of Segment IDs used for the computed path from source to destination.

Disjointness constraint works, if the Segment IDs of both the SR-TE policies are different. For example:

```
SRTE Policy 1:         SRTE Policy 2:

       SID[0]: Node, Label 16002, NAI: 207.207.207.207             SID[0]: Node, Label
16003, NAI: 208.208.208.208
       SID[1]: Node, Label 16004, NAI: 201.201.201.201           SID[1]: Node, Label 16006,
 NAI: 206.206.206.206
       SID[2]: Node, Label 16011, NAI: 205.205.205.205        SID[2]: Node, Label 16011,
NAI: 205.205.205.205
```

**Note** SID[2] of policies 1 and 2 is the same since destination of both the SR-TE policies is the same.

To view the SR-TE policy and Affinity constraint in the PCE:

```
RP/0/RSP0/CPU0:C8xxx# show pce lsp pcc ipv4 213.213.213.213 private

Thu Jan 10 00:11:52.983 UTC

PCE's tunnel database:
----------------------
PCC 213.213.213.213:
Tunnel Name: 209.209.209.209|100
 LSPs:
  LSP[0]:
   source 203.203.203.203, destination 209.209.209.209, tunnel ID 177, LSP ID 0
   State: Admin up, Operation ----□ SRTE Policy is up
   Setup type: Segment Routing
   Binding SID: 87
   Maximum SID Depth: 4
   Absolute Metric Margin: 0
   Relative Metric Margin: 0%
   Affinity: exclude-any 0x0 include-any 0x2 include-all 0x0  ----□ This indicates Affinity
 taken into account by PCE

PCEP information:
     PLSP-ID 0x800b1, flags: D:1 S:0 R:0 A:1 O:2 C:0
   LSP Role: Disjoint LSP
   State-sync PCE: None
   PCC: 213.213.213.213
   LSP is subdelegated to: None
   Reported path:
     Metric type: TE, Accumulated Metric 53
      SID[0]: Node, Label 18010, Address 202.202.202.202
      SID[1]: Node, Label 18007, Address 211.211.211.211
      SID[2]: Node, Label 18002, Address 207.207.207.207
      SID[3]: Adj, Label 21, Address: local 10.10.20.2 remote 10.10.20.1
   Computed path: (Local PCE)
     Computed Time: Thu Jan 10 00:09:36 UTC 2019 (00:02:17 ago)
     Metric type: TE, Accumulated Metric 53
      SID[0]: Node, Label 18010, Address 202.202.202.202
      SID[1]: Node, Label 18007, Address 211.211.211.211
      SID[2]: Node, Label 18002, Address 207.207.207.207
      SID[3]: Adj, Label 21, Address: local 10.10.20.2 remote 10.10.20.1
   Recorded path:
     None
   Disjoint Group Information:
     Type Node-Disjoint, Group 1, Sub-Group 1.0.0.0




Event history (latest first):
  Time                            Event
  Thu Jan 10 00:09:37 UTC 2019    Report from 213.213.213.213 (LSP owner)
                                  Symbolic-name: 209.209.209.209|100, LSP-ID: 0,
                                  Source: 203.203.203.203 Destination: 209.209.209.209,
                                  D:1, R:0, A:1 O:2, Sig.BW: 0, Act.BW: 0
                                  Reported Path: (Metric 53)
                                  Label 18010, Address 202.202.202.202
```

```
                                Label 18007, Address 211.211.211.211
                                Label 18002, Address 207.207.207.207
                                Label 21, Address: local 10.10.20.2 remote 10.10.20.1
                                Chng:0, AssoChng:0
  Thu Jan 10 00:09:36 UTC 2019  Update to 213.213.213.213 (PCC)
                                Symbolic-name: 209.209.209.209|100, LSP-ID: 0, D:1
                                Path: (Metric 53)
                                Label 18010, Address 202.202.202.202
                                Label 18007, Address 211.211.211.211
                                Label 18002, Address 207.207.207.207
                                Label 21, Address: local 10.10.20.2 remote 10.10.20.1
Thu Jan 10 00:09:36 UTC 2019  Path Computation (Disjoint LSP)
                                Symbolic-name: 209.209.209.209|100, LSP-ID: 0, D:1
                                Source: 203.203.203.203 Destination: 209.209.209.209
                                Status: Disjoint Path Success

Wed Jan 09 23:54:42 UTC 2019  Update to 213.213.213.213 (PCC)
                                Symbolic-name: 209.209.209.209|100, LSP-ID: 0, D:1
                                Path: (Metric 53)
                                Label 18007, Address 211.211.211.211
                                Label 18002, Address 207.207.207.207
                                Label 21, Address: local 10.10.20.2 remote 10.10.20.1
  Wed Jan 09 23:54:42 UTC 2019  Path Computation (Disjoint LSP)
                                Symbolic-name: 209.209.209.209|100, LSP-ID: 0, D:1
                                Source: 203.203.203.203 Destination: 209.209.209.209
                                Status: Fallback Node to Shortest Path
                                Computed Path: (Metric 53)
                                Label 18007, Address 211.211.211.211
                                Label 18002, Address 207.207.207.207
                                Label 21, Address: local 10.10.20.2 remote 10.10.20.1
  Wed Jan 09 23:54:21 UTC 2019  Path Computation (Disjoint LSP)
                                Symbolic-name: 209.209.209.209|100, LSP-ID: 0, D:1
                                Source: 203.203.203.203 Destination: 209.209.209.209
                                Status: Disjoint Path Success
Computed Path: (Metric 53)
                                Label 18010, Address 202.202.202.202
                                Label 18007, Address 211.211.211.211
                                Label 18002, Address 207.207.207.207
                                Label 21, Address: local 10.10.20.2 remote 10.10.20.1
Computed Path: (Metric 53)
                                Label 18010, Address 202.202.202.202
                                Label 18007, Address 211.211.211.211
                                Label 18002, Address 207.207.207.207
                                Label 21, Address: local 10.10.20.2 remote 10.10.20.1
  Thu Jan 10 00:09:05 UTC 2019  Path Computation (Disjoint LSP)
                                Symbolic-name: 209.209.209.209|100, LSP-ID: 0, D:1
                                Source: 203.203.203.203 Destination: 209.209.209.209
                                Status: Fallback Node to Shortest Path
                                Computed Path: (Metric 53)
                                Label 18007, Address 211.211.211.211
                                Label 18002, Address 207.207.207.207
                                Label 21, Address: local 10.10.20.2 remote 10.10.20.1
  Wed Jan 09 23:54:42 UTC 2019  Report from 213.213.213.213 (LSP owner)
                                Symbolic-name: 209.209.209.209|100, LSP-ID: 0,
                                Source: 203.203.203.203 Destination: 209.209.209.209,
                                D:1, R:0, A:1 O:2, Sig.BW: 0, Act.BW: 0
                                Reported Path: (Metric 53)
                                Label 18007, Address 211.211.211.211
                                Label 18002, Address 207.207.207.207
                                Label 21, Address: local 10.10.20.2 remote 10.10.20.1
                                Chng:0, AssoChng:0
RP/0/RSP0/CPU0:C8xxx#
```

To view dispointness between policies 1 and 2:

```
RP/0/RSP0/CPU0:C8xxx# show pce association type link group-id 3

Wed Aug 29 05:56:52.228 UTC
PCE's association database:
----------------------
Association: Type Link-Disjoint, Group 3, Sub-Group 1.0.0.0, Not Strict
 Associated LSPs:
  LSP[0]:
   PCC 213.213.213.213, tunnel name 209.209.209.209|104,  PLSP ID 524460, tunnel ID 172,
LSP ID 0, Configured on PCC
  LSP[1]:
   PCC 213.213.213.213, tunnel name 209.209.209.209|105,  PLSP ID 524461, tunnel ID 173,
LSP ID 0, Configured on PCC
 Status: Satisfied  ---------------  This indicates that Disjointness between SRTE Policies
 is working
RP/0/RSP0/CPU0:C8xxx#
```

# Troubleshooting the SR-TE ODN Color Extended Community, Affinity Constraint, and Disjointness Constraint

**Note** C8xxx=C8200/C8300/C8500 or C8000v

If SR-TE policy is down, check the status of the SR-TE Policy under the SR-TE policy information

```
C8xxx# show segment-routing traffic-eng policy name 209.209.209.209|100
Name: 209.209.209.209|106 (Color: 106 End-point: 209.209.209.209)
  Status:
    Admin: up, Operational: down for 00:00:18 (since 01-10 13:06:42.142)
  Candidate-paths:
    Preference 1:
      Constraints:
        Affinity:
          include-any:
           1
      Dynamic (pce) (inactive)
        Weight: 0, Metric Type: IGP
  Attributes:
    Binding SID: 269
      Allocation mode: dynamic
      State: Programmed
  Auto-policy info:
    Creator: BGP SR Policy Client
    IPv6 caps enable: yes
C8xxx#
```

**Note** The possible reasons for the policy being down are:

- Connection to PCE is down.
- Max SID depth is exceeded.
- An interface falling under Affinity-defined path from source to destination has been shut down.

To check the SR-TE policy status on the PCE:

```
RP/0/RSP0/CPU0:C8xxx#show pce lsp pcc ipv4 213.213.213.213 private
Thu Jan 10 00:11:52.983 UTC
PCE's tunnel database:
----------------------
PCC 213.213.213.213:
Tunnel Name: 209.209.209.209|100
 LSPs:
  LSP[0]:
   source 203.203.203.203, destination 209.209.209.209, tunnel ID 177, LSP ID 0
   State: Admin up, Operation active    -----  SRTE Policy is up
   Setup type: Segment Routing
   Binding SID: 87




 Maximum SID Depth: 4
   Absolute Metric Margin: 0
   Relative Metric Margin: 0%

Affinity: exclude-any 0x0 include-any 0x2 include-all 0x0   ---  This indicates Affinity
is taken into account by the PCE
```

PCE is aware of the network topology. This information is used for path computation using the following command. This information is also used to determine if nodes and links are present and have the expected attributes (IGP/TE admin weights, SIDs and so on).

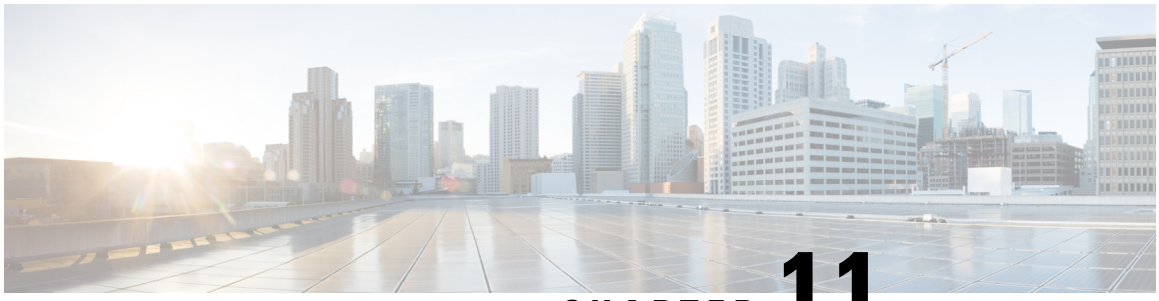
```
RP/0/RSP0/CPU0:C8xxx# show pce ipv4 topology
Tue Jan 15 01:36:20.298 UTC
PCE's topology database - detail:
---------------------------------
Node 1
  TE router ID: 207.207.207.207
  Host name: 920-R7
  ISIS system ID: 0000.0000.0207 level-1 ASN: 1
  ISIS system ID: 0000.0000.0207 level-2 ASN: 1
  Prefix SID:
 ISIS system ID: 0000.0000.0207 level-1 ASN: 1 domain ID: 0
      Prefix 207.207.207.207, label 16002 (regular), flags: N
    ISIS system ID: 0000.0000.0207 level-1 ASN: 1 domain ID: 0
      Prefix 207.207.207.207, label 18002 (strict), flags: N
ISIS system ID: 0000.0000.0207 level-2 ASN: 1 domain ID: 0
      Prefix 207.207.207.207, label 16002 (regular), flags: N
    ISIS system ID: 0000.0000.0207 level-2 ASN: 1 domain ID: 0
      Prefix 207.207.207.207, label 18002 (strict), flags: N
  SRGB INFO:
    ISIS system ID: 0000.0000.0207 level-1 ASN: 1
      SRGB Start: 16000 Size: 8000
    ISIS system ID: 0000.0000.0207 level-2 ASN: 1
      SRGB Start: 16000 Size: 8000
  Link[0]: local address 10.10.21.1, remote address 10.10.21.2
    Local node:
      ISIS system ID: 0000.0000.0207 level-1 ASN: 1
    Remote node:
      TE router ID: 208.208.208.208
      Host name: 920-R8
      ISIS system ID: 0000.0000.0208 level-1 ASN: 1
    Metric: IGP 10, TE 10, Latency 10
    Bandwidth: Total 1250000000 Bps, Reservable 0 Bps
    Admin-groups: 0x00000000
    Adj SID: 16 (unprotected) 17 (protected)
 Link[1]: local address 10.10.21.1, remote address 10.10.21.2
```

```
    Local node:
ISIS system ID: 0000.0000.0207 level-2 ASN: 1
    Remote node:
      TE router ID: 208.208.208.208
      Host name: 920-R8
      ISIS system ID: 0000.0000.0208 level-2 ASN: 1
    Metric: IGP 10, TE 10, Latency 10
    Bandwidth: Total 1250000000 Bps, Reservable 0 Bps
    Admin-groups: 0x00000000
    Adj SID: 18 (unprotected) 19 (protected)
  Link[2]: local address 10.10.20.2, remote address 10.10.20.1
    Local node:
      ISIS system ID: 0000.0000.0207 level-2 ASN: 1
    Remote node:
      TE router ID: 209.209.209.209
      Host name: 920-R9
      ISIS system ID: 0000.0000.0209 level-2 ASN: 1
    Metric: IGP 40, TE 40, Latency 40
    Bandwidth: Total 1250000000 Bps, Reservable 0 Bps
    Admin-groups: 0x00000052
    Adj SID: 20 (unprotected) 22 (protected)
    SRLG Values: 25
Node 2
  TE router ID: 209.209.209.209
  Host name: 920-R9
  ISIS system ID: 0000.0000.0209 level-1 ASN: 1
  ISIS system ID: 0000.0000.0209 level-2 ASN: 1
  Prefix SID:
    ISIS system ID: 0000.0000.0209 level-1 ASN: 1 domain ID: 0
      Prefix 209.209.209.209, label 16001 (regular), flags: N
    ISIS system ID: 0000.0000.0209 level-1 ASN: 1 domain ID: 0
      Prefix 209.209.209.209, label 18001 (strict), flags: N
    ISIS system ID: 0000.0000.0209 level-2 ASN: 1 domain ID: 0
      Prefix 209.209.209.209, label 16001 (regular), flags: N
    ISIS system ID: 0000.0000.0209 level-2 ASN: 1 domain ID: 0
      Prefix 209.209.209.209, label 18001 (strict), flags: N
  SRGB INFO:
    ISIS system ID: 0000.0000.0209 level-1 ASN: 1
      SRGB Start: 16000 Size: 8000
    ISIS system ID: 0000.0000.0209 level-2 ASN: 1
      SRGB Start: 16000 Size: 8000
 Link[0]: local address 10.10.20.1, remote address 10.10.20.2
    Local node:
 ISIS system ID: 0000.0000.0209 level-2 ASN: 1
    Remote node:
TE router ID: 207.207.207.207
      Host name: 920-R7
      ISIS system ID: 0000.0000.0207 level-2 ASN: 1
    Metric: IGP 40, TE 40, Latency 40
    Bandwidth: Total 1250000000 Bps, Reservable 0 Bps
    Admin-groups: 0x00000052
    Adj SID: 1980 (unprotected) 1981 (protected)
  Link[1]: local address 10.10.22.1, remote address 10.10.22.2
    Local node:
      ISIS system ID: 0000.0000.0209 level-2 ASN: 1
    Remote node:
      TE router ID: 208.208.208.208
      Host name: 920-R8
      ISIS system ID: 0000.0000.0208 level-2 ASN: 1
    Metric: IGP 10, TE 50, Latency 50
    Bandwidth: Total 1250000000 Bps, Reservable 0 Bps
    Admin-groups: 0x0000002C
    Adj SID: 1971 (unprotected) 1972 (protected)
```

```
RP/0/RSP0/CPU0:C8xxx#
```

Further troubleshooting tips:

- Enable the following debug commands on the PCCs:
  - debug segment-routing traffic-eng path
  - debug segment-routing traffic-eng pcalc
  - debug segment-routing traffic-eng policy
  - debug segment-routing traffic-eng topology
  - debug segment-routing traffic-eng ha
- Enable the following debug commands on the PCE:
  - debug pce pcep
  - debug pce cspf
  - debug pce cspf-internal
  - debug pce error
  - debug pce path



CHAPTER 11

# Segment Routing On Demand for L2VPN/VPWS

On-Demand Next Hop (ODN) for Layer 2 Virtual Private Network (L2VPN) creates a segment routing (SR) traffic-engineering (TE) auto-tunnel and uses the auto-tunnel for pseudowire dataplane.


- Feature Information for Segment Routing On Demand Next Hop for L2VPN/VPWS, on page 135
- Restrictions for Segment Routing On Demand Next Hop for L2VPN/VPWS, on page 136
- Information About Segment Routing On Demand Next Hop for L2VPN/VPWS, on page 136
- How to Configure Segment Routing On Demand Next Hop for L2VPN/VPWS, on page 137
- Configuring Segment Routing On Demand Next Hop for L2VPN/VPWS With Prepend Option, on page 139
- Configuring Preferred Path for Segment Routing On Demand Next Hop for L2VPN/VPWS, on page 139
- Configuring Autoroute Destination for Segment Routing On Demand Next Hop for L2VPN/VPWS, on page 140
- Verifying Segment Routing On Demand Next Hop for L2VPN/VPWS , on page 140


# Feature Information for Segment Routing On Demand Next Hop for L2VPN/VPWS

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

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

*Table 10: Feature Information for Segment Routing On Demand Next Hop for L2VPN/VPWS*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing On Demand Next Hop for L2VPN/VPWS | Cisco IOS XEAmsterdam 17.3.2 | ODN for L2VPN is to create a SR TE auto-tunnel and use the auto-tunnel for pseudo-wire data-plane. The peer IP address is the destination of tunnel and TE LSP attribute determines the path of the tunnel.<br>The following commands were added or modified:<br>**sh mpls l2 vc**, **sh mpls l2 vc detail**, **sh l2vpn atom preferred-path**, **sh l2vpn atom vc**, **sh mpl traffic-eng tun tun 2000**, **sh mpls ldp discovery**, **sh mpl ldp nei**, **sh int pseudowire 4243**, **sh xconnect all**. |

# Restrictions for Segment Routing On Demand Next Hop for L2VPN/VPWS

- Layer-2 VPN/VPWS (Virtual Private Wire Service) On Demand Next Hop (ODN) is not supported with peudowire (PW) class.
- The segment routing on demand for L2VPN or VPWS is not supported for BGP signaled/ADVPWS or Virtual Private LAN Service (VPLS).
- Only Segment-Routing TE tunnels are supported and created for L2VPN using attribute-set.
- L2VPN preferred path bandwidth related configuration does not take effect when TE attribute-set is configured.
- Only L2-VPN ODN VPWS with LDP signaling is supported.

# Information About Segment Routing On Demand Next Hop for L2VPN/VPWS

On Demand Next Hop (ODN) for L2VPN creates an SR TE auto-tunnel and uses the auto-tunnel for pseudowire dataplane. The peer IP address is the destination of tunnel and TE LSP attribute determines path of the tunnel. Sometimes a pseudowire connection may need to span multiple interior gateway protocol (IGP) areas while LDP is used as signaling protocol. The pseudowire endpoint provider edge's (PE) loopback addresses are not distributed across IGP area boundaries. In this case, one PE may not have a default route (or an exact match route) in its RIB to reach the peer PE of the pseudowire connection. Thus the pseudowire connection can not be signaled by LDP. A new option **autoroute destination** is introduced under LSP attribute to address this problem. When a LSP attribute is configured using the **autoroute destination** command, auto-tunnel uses the LSP attribute to automatically create a static route for the tunnel destination with the auto-tunnel interface as the next hop. This static route enables LDP to establish a LDP a session and exchange label mapping messages between two pseudowire endpoints.

**Note** Use the autoroute destination command only to configure LSP attribute used by LDP signaled L2VPN. It is not needed for BGP signaled Layer-3 VPN ODN.

## AToM Manager

Any Transport over MPLS (AToM) manager maintains a database of auto-tunnels on a pair of attribute set and peer ip addresses, the AToM manager can add or delete an SR TE auto-tunnel for a pseudowire interface (VC).

Any VC that is configured with the same attribute-set or peer uses the same auto-tunnel. An auto-tunnel can be removed from the database using TE service if an attribute set or peer pair is no longer used by any pseudowire interfaces.

## Inter-Area L2VPN ODN

When LDP is used as a signaling protocol and pesudowire connection is spanned across multiple Interior Gateway Protocols (IGPs), the pseudowire endpoint PE's loopback addresses are not distributed across IGP area boundaries. In this case, one PE may not have a default route (or an exact match route) in its RIB to reach the peer PE of the pseudowire connection. Thus the pseudowire connection can not be signaled by LDP.

# How to Configure Segment Routing On Demand Next Hop for L2VPN/VPWS

You can use either pesudowire interface command or template method to configure L2VPN/VPWS.

## Configuring Segment Routing On Demand Next Hop for L2VPN/VPWS Using Pesudowire Interface Commands

1. Run the following command on headend node (R1):

```
R1#
!
mpls traffic-eng auto-tunnel p2p tunnel-num min 2000 max 2002
!
interface GigabitEthernet0/3/1
 no ip address
 negotiation auto
 service instance 300 ethernet
  encapsulation dot1q 300
!
interface pseudowire4243
 encapsulation mpls
 neighbor 56.6.6.6 300
 preferred-path segment-routing traffic-eng attribute-set L2VPNODN
!
l2vpn xconnect context foobar
 member GigabitEthernet0/3/1 service-instance 300
```

```
 member pseudowire4243
!
mpls traffic-eng lsp attributes L2VPNODN
 priority 7 7
 path-selection metric te
!
end
```

2. Run the following command at tail end (R2):

```
R2#
!
mpls traffic-eng auto-tunnel p2p tunnel-num min 2000 max 2002

interface pseudowire4243
 encapsulation mpls
 neighbor 51.1.1.1 300
 preferred-path segment-routing traffic-eng attribute-set L2VPNODN
!
interface GigabitEthernet0/2/2
 no ip address
 negotiation auto
 service instance 300 ethernet
  encapsulation dot1q 300
!
l2vpn xconnect context foobar
 member GigabitEthernet0/3/1 service-instance 300
 member pseudowire4243
!
mpls traffic-eng lsp attributes L2VPNODN
 priority 7 7
 path-selection metric te
!
end
```

# Configuring Segment Routing On Demand Next Hop for L2VPN/VPWS Using Template Commands

1. Run the following command at headend node (R1):

```
R1#
template type pseudowire test
 encapsulation mpls
 preferred-path segment-routing traffic-eng attribute-set L2VPNODN
!
interface GigabitEthernet0/3/1
 no ip address
 negotiation auto
 service instance 400 ethernet
  encapsulation dot1q 400
!
l2vpn xconnect context foobar2
 member 56.6.6.6 400 template test
 member GigabitEthernet0/3/1 service-instance 400
```

2. Run the following command at tail end (R2):

```
R2#
!
```

```
template type pseudowire test
 encapsulation mpls
 preferred-path segment-routing traffic-eng attribute-set L2VPNODN
!
interface GigabitEthernet0/2/2
 no ip address
 negotiation auto
 service instance 400 ethernet
  encapsulation dot1q 400
!
l2vpn xconnect context foobar2
 member 51.1.1.1 400 template test
 member GigabitEthernet0/2/2 service-instance 400
!
end
```

# Configuring Segment Routing On Demand Next Hop for L2VPN/VPWS With Prepend Option

To control the path of LSP it is possible to enable prepend option. The prepend option is only supported with intra-area and supports labeled paths only. To enable prepend option use the following CLI:

```
R1(config-lsp-attr)#path-selection segment-routing prepend
R1(config-lsp-attr-sr-prepend)#?
Segment-routing label prepend commands:
  exit   Exist from segment-routing prepend config mode
  index  Specify the next entry index to add, edit or delete
  list   List all prepend entries
  no     Delete a specific entry index
R1(config-lsp-attr-sr-prepend)#index ?
  <1-10>  Entry index number
  last-hop    Indicates the end of label list
  next-label  Specify the next MPLS label in the path
```

**Note** If last-hop option indicates tail end node. If this option is only used no control on LSP path can be done.

# Configuring Preferred Path for Segment Routing On Demand Next Hop for L2VPN/VPWS

To bring down virtual circuit (VC) in case of LSP failure, which could be either because of path fail or removing a command, disable the fallback mode.

```
preferred-path segment-routing traffic-eng attribute-set L2VPNODN
disable-fallback  disable fall back to alternative route
```

# Configuring Autoroute Destination for Segment Routing On Demand Next Hop for L2VPN/VPWS

For inter-area destination, IP address may not be installed at headend. You need to have destination IP address installed to enable a targeted LDP session for L2-VPN VPWS. To enable a targeted LDP session for L2VPN VPWS, configure the auto-route destination under the attribute set:

```
Device#
mpls traffic-eng lsp attributes L2VPNODN
 priority 7 7
 path-selection metric te
 pce
 autoroute destination
!
end
```

The destination address gets installed via L2-VPN ODN LSP as a static route.

Run the following commands to verify autoroute destination configuration:

```
Device#sh ip route 56.6.6.6
Routing entry for 56.6.6.6/32
  Known via "static", distance 1, metric 0 (connected)
  Routing Descriptor Blocks:
  * directly connected, via Tunnel2000------------------------ L2-VPN ODN LSP
      Route metric is 0, traffic share count is 1
```

```
Device#sh mpls for 56.6.6.6
Local      Outgoing             Prefix          Bytes Label      Outgoing   Next Hop
Label      Label               or Tunnel Id     Switched          interface
25           [T]  Pop Label    56.6.6.6/32      0                   Tu2000     point2point
```

# Verifying Segment Routing On Demand Next Hop for L2VPN/VPWS

1. **sh mpls l2 vc**

```
Device#sh mpls l2 vc
Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
Gi0/3/1        Eth VLAN 300               56.6.6.6        300        UP
```

2. **sh mpls l2 vc detail**

```
Device# sh mpls l2 vc detail
Local interface: Gi0/3/1 up, line protocol up, Eth VLAN 300 up
  Interworking type is Ethernet
  Destination address: 56.6.6.6, VC ID: 300, VC status: up
    Output interface: Tu2000, imposed label stack {23 17 20}----- 20 is the VC label
assigned by R6
    Preferred path: Tunnel2000,  active
    Default path: ready
```

```
    Next hop: point2point
  Create time: 00:15:48, last status change time: 00:15:38
    Last label FSM state change time: 00:15:38
  Signaling protocol: LDP, peer 56.6.6.6:0 up
    Targeted Hello: 51.1.1.1(LDP Id) -> 56.6.6.6, LDP is UP
    Graceful restart: not configured and not enabled
    Non stop routing: not configured and not enabled
    Status TLV support (local/remote)   : enabled/supported
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: No fault
      Last BFD dataplane     status rcvd: Not sent
      Last BFD peer monitor  status rcvd: No fault
      Last local AC  circuit status rcvd: No fault
      Last local AC  circuit status sent: No fault
      Last local PW i/f circ status rcvd: No fault
      Last local LDP TLV     status sent: No fault
      Last remote LDP TLV    status rcvd: No fault
      Last remote LDP ADJ    status rcvd: No fault
    MPLS VC labels: local 2032, remote 20
    Group ID: local 20, remote 25
    MTU: local 1500, remote 1500
    Remote interface description:
  Sequencing: receive disabled, send disabled
  Control Word: On (configured: autosense)
  SSO Descriptor: 56.6.6.6/300, local label: 2032
  Dataplane:
    SSM segment/switch IDs: 10198/6097 (used), PWID: 1001
  VC statistics:
    transit packet totals: receive 0, send 0
    transit byte totals:   receive 0, send 0
    transit packet drops:  receive 0, seq error 0, send 0
```

### 3. sh l2vpn atom preferred-path

```
Device#  sh l2vpn atom preferred-path
Tunnel interface    Bandwidth Tot/Avail/Resv         Peer ID         VC ID
------------------       ---------------------------------          ---------------
 ----------
Tunnel2000                                                             56.6.6.6
 300
!
end
```

### 4. sh l2vpn atom vc

```
Device# sh l2vpn atom vc
Interface Peer ID         VC ID      Type   Name                       Status
--------- --------------      ----------    ------ ------------------------ ----------
pw4243    56.6.6.6        300        p2p    foobar                          UP
!
end
```

### 5. sh mpl traffic-eng tun tun 2000

```
Device# sh mpl traffic-eng tun tun 2000
Name: R1_t2000                 (Tunnel2000) Destination: 56.6.6.6 Ifhandle: 0x7EE
(auto-tunnel for atom)
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 1, (SEGMENT-ROUTING) (PCE) type dynamic (Basis for Setup, path weight
```

```
30)
  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF
    Metric Type: TE (interface)
    Path Selection:
     Protection: any (default)
    Path-selection Tiebreaker:
      Global: not set   Tunnel Specific: not set   Effective: min-fill (default)
    Hop Limit: disabled
    Cost Limit: disabled
    Path-invalidation timeout: 10000 msec (default), Action: Tear
    AutoRoute: disabled LockDown: disabled Loadshare: 0 [0] bw-based
    auto-bw: disabled
    Attribute-set: L2VPNODN
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: dynamic path option 1 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
PCEP Info:
    Delegation state: Working: yes   Protect: no
    Delegation peer: 58.8.8.8
    Working Path Info:
      Request status: processed
      Created via PCRep message from PCE server: 58.8.8.8
      PCE metric: 30, type: TE
    Reported paths:
      Tunnel Name: Tunnel2000_w
       LSPs:
        LSP[0]:
         source 51.1.1.1, destination 56.6.6.6, tunnel ID 2000, LSP ID 4
         State: Admin up, Operation active
         Binding SID: 20
         Setup type: SR
         Bandwidth: requested 0, used 0
         LSP object:
           PLSP-ID 0x807D0, flags: D:0 S:0 R:0 A:1 O:2
         Metric type: TE, Accumulated Metric 30
         ERO:
           SID[0]: Adj, Label 19, NAI: local 101.104.1.1 remote 101.104.1.2
           SID[1]: Adj, Label 23, NAI: local 103.104.12.2 remote 103.104.12.1
           SID[2]: Adj, Label 17, NAI: local 103.106.13.1 remote 103.106.13.2
       PLSP Event History (most recent first):
        Tue Jun 20 10:04:48.514: PCRpt create LSP-ID:4, SRP-ID:0, PST:1, METRIC_TYPE:2,
 REQ_BW:0, USED_BW:0
        Tue Jun 20 10:04:48.511: PCRep RP-ID:9
        Tue Jun 20 10:04:48.505: PCReq RP-ID:9, LSP-ID:4, REQ_BW:0
  History:
    Tunnel:
      Time since created: 18 minutes, 26 seconds
      Time since path change: 17 minutes, 9 seconds
      Number of LSP IDs (Tun_Instances) used: 4
    Current LSP: [ID: 4]
      Uptime: 17 minutes, 9 seconds
  Tun_Instance: 4
  Segment-Routing Path Info (isis  level-2)
    Segment0[Link]: 101.104.1.1 - 101.104.1.2, Label:  19--------□ will not be shown in
 sh mpls l2 vc output
    Segment1[Link]: 103.104.12.2 - 103.104.12.1, Label: 23
    Segment2[Link]: 103.106.13.1 - 103.106.13.2, Label: 17
!
end
```

6. **sh mpls ldp discovery**

```
Device# sh mpls ldp discovery
Local LDP Identifier:
    51.1.1.1:0
    Discovery Sources:
    Targeted Hellos:
        51.1.1.1 -> 56.6.6.6 (ldp): active/passive, xmit/recv
            LDP Id: 56.6.6.6:0
```

7. **sh mpl ldp nei**

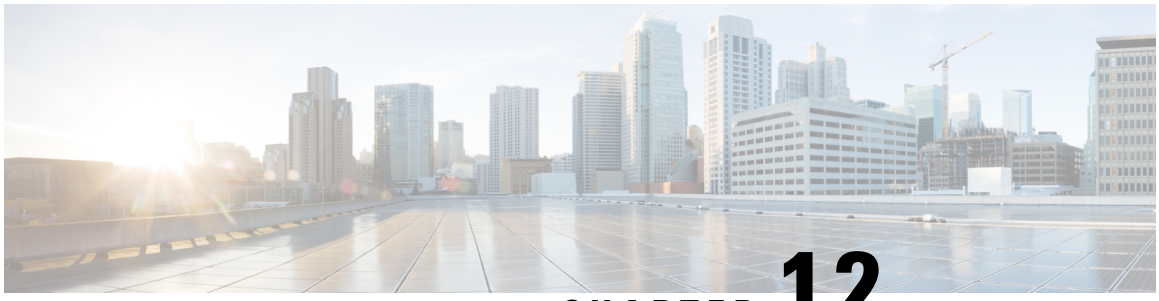
```
Device# sh mpl ldp nei
Peer LDP Ident: 56.6.6.6:0; Local LDP Ident 51.1.1.1:0
        TCP connection: 56.6.6.6.38574 - 51.1.1.1.646
        State: Oper; Msgs sent/rcvd: 43/42; Downstream
        Up time: 00:19:33
        LDP discovery sources:
          Targeted Hello 51.1.1.1 -> 56.6.6.6, active, passive
        Addresses bound to peer LDP Ident:
          105.106.2.2     103.106.13.2    56.6.6.6
!
```

8. **sh int pseudowire 4243**

```
Device# sh int pseudowire 4243
pseudowire4243 is up
    MTU 1500 bytes, BW not configured
    Encapsulation mpls
    Peer IP 56.6.6.6, VC ID 300
    RX    0 packets 0 bytes 0 drops
    TX    0 packets 0 bytes 0 drops
!
```

9. **sh xconnect all**

```
Device# sh xconnect all
Legend:    XC ST=Xconnect State  S1=Segment1 State  S2=Segment2 State
  UP=Up       DN=Down            AD=Admin Down      IA=Inactive
  SB=Standby  HS=Hot Standby     RV=Recovering      NH=No Hardware

XC ST  Segment 1                                    S1 Segment 2
  S2
------+---------------------------------+--+---------------------------------+--
UP pri   ac Gi0/3/1:300(Eth VLAN)        UP mpls 56.6.6.6:300                 UP
```



CHAPTER 12

# Fast Convergence Default Optimize

The fast convergence default optimize feature modifies the default settings of all the protocols to recommended defaults for fast convergence.


- Feature Information for Fast Convergence Default Optimize, on page 145
- Information About Fast Convergence Default Optimize, on page 145


## Feature Information for Fast Convergence Default Optimize

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

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

***Table 11: Feature Information for Fast Convergence Default Optimize***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Fast Convergence Default Optimize | Cisco IOS XE Amsterdam 17.3.2 | The fast convergence default optimize feature modifies the default settings of all the protocols to recommended defaults for fast convergence.<br>No new commands were added or modified. |

## Information About Fast Convergence Default Optimize

The fast convergence default optimize feature modifies the default settings of all the protocols to recommended defaults for fast convergence. To revert the defaults to pre-fast-convergence settings for both IS-IS and OSPF, **no routing-default-optimize** command is used. This command sends signals to IS-IS and OSPF and modifies the default configuration for these protocols.

By default, the fast convergence settings is enabled which means when you upgrade the software, you can automatically see the new behavior. This makes easier integration of the devices in a multi-vendor deployment and reduces support cases for poor convergence.

When default optimize is disabled, existing protocol default configuration is used. When default optimize is enabled, new protocol defaults are used. The show running configurations does not display configuration lines for default settings even when default settings are being used.

A configuration of a protocol overrides the default, but a change to default optimize does not override any configuration.

The following is the sample output of **spf-interval** command in IS-IS:

```
Device(config-if)# router isis
Device(config-router)# spf-interval 10 5500 5500
```

If a non-default value is configured, it will be displayed in show running configuration output:

```
Device(config-router)# spf-interval 5 50 200
Device(config-router)# do show run | inc spf-interval
 spf-interval 5 50 200
```

You can revert to the default values by configuring the default values or by removing the non-default configuration.

# Default Optimize Values for IS-IS

The following table summarizes the configuration impacted by default optimize:

| IS-IS command | Parameters | Default optimize disabled | Default optimize enabled |
|---|---|---|---|
| fast-flood | | | |
| | # of lsps flooded back-back | Disabled | 10 |
| spf-interval | | | |
| | Initial (milliseconds) | 5500 | 50 |
| | Secondary (milliseconds) | 5500 | 200 |
| | max (seconds) | 10 | 5 |
| prc-interval | | | |
| | Initial (milliseconds) | 2000 | 50 |
| | Secondary (milliseconds) | 5000 | 200 |
| | max (seconds) | 5 | 5 |
| lsp-gen-interval | | | |
| | Initial (milliseconds) | 50 | 50 |
| | Secondary (milliseconds) | 5000 | 200 |
| | max (seconds) | 5 | 5 |

| IS-IS command | Parameters | Default optimize disabled | Default optimize enabled |
|---|---|---|---|
| log-adjacency-changes | | disabled | enabled |

# Default Optimize Values for OSPF

The following table summarizes the configuration impacted by default optimize for OSPFv2/v3:

| OSPF command | Parameters | Default optimize disabled | Default optimize enabled |
|---|---|---|---|
| timers throttle spf | | | |
| | Initial (milliseconds) | 5000 | 50 |
| | Secondary (milliseconds) | 10000 | 200 |
| | max (milliseconds) | 10 | 5 |
| timers throttle lsa all | | | |
| | Initial (milliseconds) | 0 | 50 |
| | Secondary (milliseconds) | 5000 | 200 |
| | max (milliseconds) | 5 | 5 |
| timers lsa arrival | | | |
| | milliseconds | 1000 | 100 |

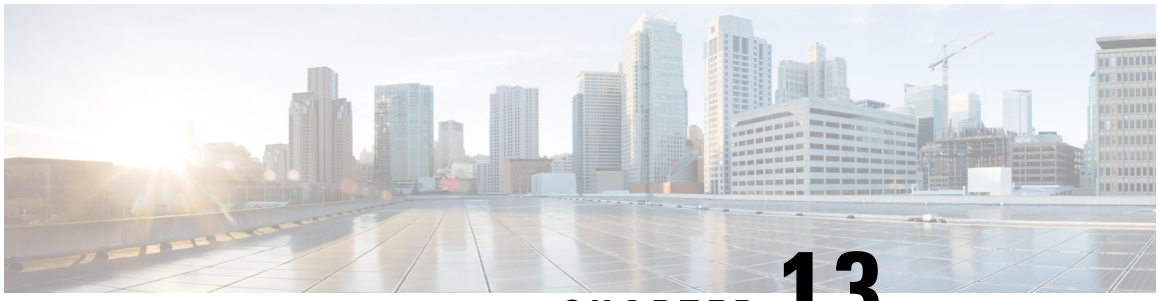
The following is the sample output of **show ip ospf** command for OSPFv2 with the default-optimize values.

```
Device# show ip ospf
 Routing Process "ospf 10" with ID 1.1.1.1
 Start time: 00:00:01.471, Time elapsed: 03:00:34.706
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101)
 Supports Database Exchange Summary List Optimization (RFC 5243)
 Event-log enabled, Maximum number of events: 1000, Mode: cyclic
 Router is not originating router-LSAs with maximum metric
 Initial SPF schedule delay 50 msecs
 Minimum hold time between two consecutive SPFs 200 msecs
 Maximum wait time between two consecutive SPFs 5000 msecs
 Incremental-SPF disabled
 Initial LSA throttle delay 50 msecs
 Minimum hold time for LSA throttle 200 msecs
 Maximum wait time for LSA throttle 5000 msecs
Minimum LSA arrival 100 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
EXCHANGE/LOADING adjacency limit: initial 300, process maximum 300
Number of external LSA 18. Checksum Sum 0x075EB2
Number of opaque AS LSA 0. Checksum Sum 0x000000
```

```
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
    Area BACKBONE(0)
        Number of interfaces in this area is 4 (2 loopback)
        Area has RRR enabled
        Area has no authentication
        SPF algorithm last executed 02:27:23.736 ago
        SPF algorithm executed 20 times
        Area ranges are
        Number of LSA 94. Checksum Sum 0x321DCF
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

The following is the sample output of **show ospf** command for OSPFv3 with the default-optimize values.

```
Device# show ospfv3
 OSPFv3 10 address-family ipv6
 Router ID 11.11.11.11
 Supports NSSA (compatible with RFC 3101)
Supports Database Exchange Summary List Optimization (RFC 5243)
Event-log enabled, Maximum number of events: 1000, Mode: cyclic
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 50 msecs
Minimum hold time between two consecutive SPFs 200 msecs
Maximum wait time between two consecutive SPFs 5000 msecs
Initial LSA throttle delay 50 msecs
Minimum hold time for LSA throttle 200 msecs
Maximum wait time for LSA throttle 5000 msecs
Minimum LSA arrival 100 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Retransmission limit dc 24 non-dc 24
EXCHANGE/LOADING adjacency limit: initial 300, process maximum 300
Number of external LSA 0. Checksum Sum 0x000000
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Graceful restart helper support enabled
Reference bandwidth unit is 100 mbps
RFC1583 compatibility enabled
    Area BACKBONE(0)
        Number of interfaces in this area is 2
        SPF algorithm executed 7 times
        Number of LSA 3. Checksum Sum 0x012426
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```



CHAPTER 13

# Routing Information Base Support

The Routing Information Base (RIB) enhancement supports route redistribution and on-demand nexthop requirements.


- Feature Information for Routing Information Base Support, on page 149
- Routing Information Base Support for Route Redistribution, on page 150
- OSPF Node SID Redistribution Support , on page 150
- Routing Information Base Support for On-Demand Next Hop, on page 152


# Feature Information for Routing Information Base Support

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

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

***Table 12: Feature Information for Routing Information Base Support***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Routing Information Base Support | Cisco IOS XE Amsterdam 17.3.2 | The Routing Information Base (RIB) enhancement supports route redistribution and On-Demand Nexthop requirements.<br><br>No new commands were added or modified. |
| OSPF Node SID Redistribution Support | Cisco IOS XE Amsterdam 17.3.2 | When OSPF receives the redistributed prefixes from other IGPs and vice versa the prefix segment identifiers (SIDs) are also advertised which was not the case earlier. You need to have the BGP LS (or) segment routing mapping server (SRMS) support to learn the SIDs across the IGP domains.<br><br>The following commands were added or modified for this feature: **show ip ospf rib redistribution detail**, **show ip ospf segment-routing local-prefix**, **show ip ospf segment-routing sid-database**, **show ip route 3.3.3.3**. |

# Routing Information Base Support for Route Redistribution

Effective with Cisco IOS XE Everest 16.5.1, a requirement to redistribute labels associated with prefixes is introduced. To support redistribution requirements, the storage of local label per prefix is supported in RIB.

The local label is stored instead of the SID to ease use with different protocols which may use different SRGBs. The SID assigned by the destination protocol may not be the same as the SID associated with the source protocol.

The prefix reachability advertisement or an SRMS advertisement is the source of the SID. In SRMS advertisement, the destination protocols for redistribution do not advertise the SID in their prefix reachability advertisements, as this alters conflict resolution by indicating on other network nodes that the source of the advertisement was not from SRMS.

# OSPF Node SID Redistribution Support

Effective Cisco IOS XE 16.7.1, when OSPF receives the redistributed prefixes from other IGPs and vice versa the prefix segment identifiers (SIDs) are also advertised which was not the case earlier. You needed to have the BGP LS (or) segment routing mapping server (SRMS) support to learn the SIDs across the IGP domains.

When the user enable redistribution under OSPF the prefix SID entries associated with the prefix entries are provided to OSPF. This gets advertised by OSPF to all its neighbor. The way OSPF advertises varies depending upon the role of OSPF in the network.

## Information About OSPF Node SID Redistribution Support

### NSSA ASBR

When you enable **redistribute ISIS** *instance* **ip** under OSPF which is Not-So-Stubby Area autonomous system boundary router (NSSA ASBR), it gets all the prefixes from IP routing information base (RIB) which are learnt by IS-IS along with the SID entries. OSPF generates Extended Prefix LSA (EPL) with the scope as area and the route type as RTYPE_NSSA1 or RTYPE_NSSA2 for the prefixes and advertises to all its neighbors. Similarly, when the redistribution is un-configured (or) when the prefixes become unavailable OSPF withdraws the EPL. When the redistributed route is a non-connected route then the OSPF sets the No-PHP flag but explicit NULL flag is not set. However, when the redistributed route is a connected route then OSPF sets the explicit NULL and No-PHP flag according to the configuration done in the SR policy.

When NSSA ABR receives the EPL, the ABR translates the LSA into opaque AS EPL and floods it to all its neighbors.

When a NSSA router which is neither ABR nor ASBR receives the EPL, it learns the prefix along with the SID entries and floods it to all its neighbors in the same area.

### non-NSSA ASBR

When the user enabled **redistribute ISIS** *instance* **ip** under OSPF which is regular ASBR router, it gets all the prefixes from IP RIB which are learnt by IS-IS along with the SID entries. OSPF generates EPL with the scope as autonomous system (AS) and the route type as RTYPE_EXTERN1 or RTYPE_EXTERN2 for the prefixes and advertises to all its neighbors. Similarly when the redistribution is unconfigured (or) when the prefixes become unavailable, OSPF withdraws the EPL again with AS-Scope. When the redistributed route

is a non-connected route then the OSPF sets the No-PHP flag but explicit NULL flag is not set. However, when the redistributed route is a connected route then OSPF sets the explicit NULL and No-PHP flag according to the configuration done in the SR policy. When a router receives the EPL with AS scope, it learns the prefix along with the SID entry and floods it to all its neighbors in all areas.

## Redistributing Prefix

When IS-IS is enabled for redistribution of OSPF routes the prefixes are given along with the SID information so that the prefixes reach to other domain with the SID values. Refer to the below topology to understand the OSPF prefixes redistribution to the other domains:

***Figure 17: OSPF Prefix Redistribution***


R1 and R2 are enabled for OSPF. R2 and R3 are enabled for IS-IS. Both IS-IS and OSPF are enabled for Segment Routing. In R2, both IS-IS and OSPF are configured. Prefixes configured are:

1. 1.1.1/32 in R1 (enabled for OSPF with SID 1)
2. 2.2.2/32 in R2 (enabled for OSPF with SID 2)
3. 3.3.3/32 in R3 (enabled for ISIS SID 3)

When you enable SID redistribution in R2, then the prefix 3.3.3.3/32is redistributed to R1. So, R1 knows the SID to reach the prefix R3.

```
conf trouter isis 10 net 49.0001.0000.0000.0001.00 metric-style wide distribute link-state
 segment-routing mpls router ospf 10 router-id 2.2.2.2 segment-routing mpls distribute
link-state
```

To enable redistribution of ISIS into OSPF routes:

```
conf t router ospf 10 redistribute isis 10 ip
```

# Verify OSPF Node SID Redistribution

Use the **show ip ospf rib redistribution detail** command to verify if OSPF is redistributing the prefixes from IS-IS.

**Note** C8xxx=C8200/C8300/C8500 or C8000v

```
c8xxx# show ip ospf rib redistribution  detail
OSPF Router with ID (2.2.2.2) (Process ID 10)

                Base Topology (MTID 0)

OSPF Redistribution
3.3.3.3/32, type 2, metric 20, tag 0, from IS-IS Router
  Attributes 0x1000000, event 1, PDB Index 4, PDB Mask 0x0
  Source route metric 20, tag 0
  SID 1003, SID Flags NP-bit, EPX Flags None
   via 7.9.0.9, Ethernet0/0
```

Use the **show ip ospf segment-routing local-prefix** command to verify if the SID entries are advertised to its neighbor.

```
c8xxx# show ip ospf segment-routing local-prefix

            OSPF Router with ID (2.2.2.2) (Process ID 10)
Area 0:
 Prefix:            Sid:   Index:          Type:      Source:
2.2.2.2/32         2      0.0.0.0         Intra      Loopback0
AS external:
Prefix:            Sid:   Index:          Type:      Source:
 3.3.3.3/32        3     0.0.0.1         External   Redist
```

Use the **show ip ospf segment-routing sid-database** command to verify if the SIDs are received.

```
Device# show ip ospf segment-routing sid-database

            OSPF Router with ID (1.1.1.1) (Process ID 10)
OSPF Segment Routing SIDs

Codes: L - local, N - label not programmed,
       M - mapping-server

SID             Prefix              Adv-Rtr-Id       Area-Id  Type
--------------  ------------------  ---------------  -------  ----------
1               1.1.1.1/32          1.1.1.1          0        Intra
2               2.2.2.2/32          2.2.2.2          0        Intra
3               3.3.3.3/32          2.2.2.2          -        External
```

Use the **show ip route 3.3.3.3** command to verify if the IP routing entry is configured for the redistributed route.

```
c8xxx# show ip route 3.3.3.3
Routing entry for 3.3.3.3/32
  Known via "ospf 10", distance 110, metric 20, type extern 2, forward metric 20
  Last update from 1.2.0.2 on Ethernet0/1, 00:00:01 ago
 SR Incoming Label: 16003
  Routing Descriptor Blocks:
  * 1.3.1.3, from 2.2.2.2, 00:00:01 ago, via Ethernet1/1, merge-labels
      Route metric is 20, traffic share count is 1
      MPLS label: 16003
      MPLS Flags: NSF
```

# Routing Information Base Support for On-Demand Next Hop

For On-Demand Next Hop (ODN) requirements, RIB supports a next hop called binding label which is provided by the supporting routing protocol (BGP). The binding label is used by the FIB to dynamically resolve the next hop.

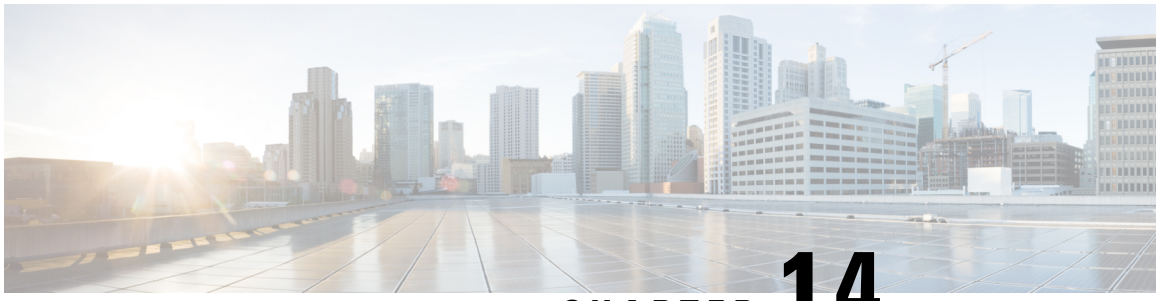
The route producer installs a local binding label which identifies the ODN tunnel path associated with the next hop. The labeled traffic is sent via the tunnel and the label is distinct from the existing outlabel.

The following is the sample output of **show ip route** command where each next hop is updated to show the binding label.

```
Device# show ip route 10.10.10.2

Routing entry for 10.10.10.2/32
```

```
Known via "isis", distance 115, metric 10, type level-1
Redistributing via isis
Last update from 200.200.200.2 on Ethernet0/0, 00:00:14 ago
Incoming Label: 16100
Routing Descriptor Blocks:
* 200.200.200.2, from 10.10.10.2, 00:00:14 ago, via Ethernet0/0
    Route metric is 10, traffic share count is 1
    * Binding Label 4020, from 2.2.2.2, 00:00:14 ago,
    Route metric is 10, traffic share count is 1
```

**Note** The incoming labels are seen only after the SID redistribution is enabled.



CHAPTER 14

# SR-TE On Demand LSP

The SR TE On demand LSP feature provides the ability to connect Metro access rings via a static route to the destination. The static route is mapped to an explicit path and that will trigger an on demand LSP to the destination. The SR TE On demand LSP feature will be used to transport the VPN services between the Metro access rings.


- Feature Information for SR-TE On Demand LSP, on page 155
- Restrictions for SR-TE On Demand LSP, on page 155
- Information About SR-TE On Demand LSP, on page 156
- How to Configure SR-TE On Demand LSP, on page 157


## Feature Information for SR-TE On Demand LSP

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

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

***Table 13: Feature Information for SR-TE On Demand LSP***

| Feature Name | Releases | Feature Information |
|---|---|---|
| SR-TE On Demand LSP | Cisco IOS XE Amsterdam 17.3.2 | The SR TE On demand LSP feature provides the ability to connect Metro access rings via a static route to the destination. The static route is mapped to an explicit path and that will trigger an on demand LSP to the destination. The SR TE On demand LSP feature will be used to transport the VPN services between the Metro access rings.<br>The following command was modified: **mpls traffic-eng auto-tunnel**. |

## Restrictions for SR-TE On Demand LSP

- Segment-Routing auto tunnel static route does not support ECMP.

- Metrics for IP explicit path and administrtive distance change for auto tunnel SRTE static route is not supported.
- MPLS Traffic Engineering (TE) Nonstop Routing (NSR) must be configured on the active route processor (RP) for Stateful Switchover (SSO). This is because, SR static auto tunnel will fail to come up after SSO, unless the static route auto tunnel configuration is removed and reconfigured.
- IP unnumbered interfaces do not support dynamic path.
- When using IP unnumbered interfaces, you cannot specify next hop address as an explicit path index. It should be a node address or a label.

# Information About SR-TE On Demand LSP

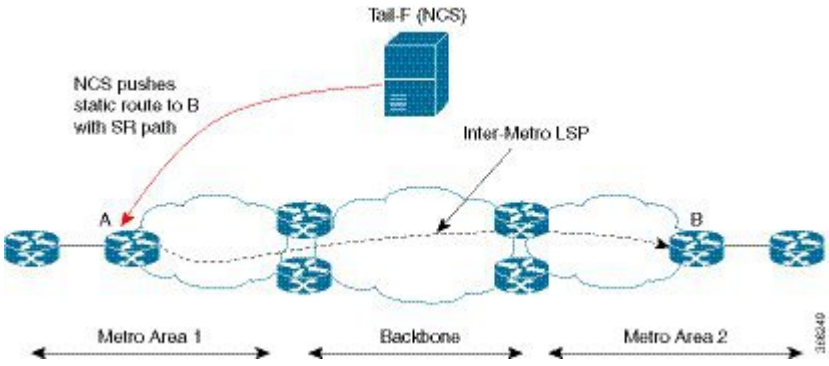
The SR TE On demand LSP feature provides the ability to connect Metro access rings via a static route to the destination.

## SR-TE: Setup LSP as Static Route

Agile Carrier Ethernet (ACE) solution leverages Segment Routing-based transport for consolidated VPN services. In metro rings architecture, the access rings do not share their routing topologies with each other.

The SR TE On demand LSP feature provides the ability to connect Metro access rings via a static route to the destination. The static route is mapped to an explicit path and that will trigger an on demand LSP to the destination. The SR TE On demand LSP feature will be used to transport the VPN services between the Metro access rings.

***Figure 18: Inter-Metro LSP in ACE Solution***




Inter-Metro LSPs have the following aspects:

- The source packet may not know the IP address of the destination device.
- Existing segment routing features are applicable for LSPs.

The binding SID helps in steering the traffic in the SR-TE tunnel. In other words, ingress MPLS packet with the binding SID will be forwarded through the specific SR-TE tunnel.

## Static SRTE over Unnumbered Interfaces

As explained in the previous section, you can set up LSP as static route to create an auto tunnel by specifying an IP explicit path.

The explicit path is a combination of IP addresses (or) IP address and labels. You can also configure the static SRTE tunnel over unnumbered interfaces. There are few restrictions for unnumbered interfaces against numbered interfaces.

- You must specify the node IP address, not the next hop interface address in the ip-explicit path option.
- You must not specify adjacency SID in the explicit path option. In short, the explicit path option should contain only the node IP address (/32 mask) and prefix SID labels.

# How to Configure SR-TE On Demand LSP

Perform the following steps to configure SR-TE On Demand LSP.

## Configuring LSP as Static Route

To avoid packet drop after RP switchover with SR TE, it is recommended to use the following command:

```
mpls traffic-eng nsr
```

If ISIS is configured, use the following command:

```
router isis
 nsf cisco
 nsf interval 0
```

## Enabling Segment Routing Auto Tunnel Static Route

Perform this task to configure auto tunnel static route as follows:

- Configure IP explicit path
- Associate the auto tunnel with an IP explicit path with a static route
- Enable peer-to-peer (P2P) auto tunnel service

```
ip explicit-path name path1
 index 1 next-label 16002
 index 2 next-label 16006
 exit
ip route 172.16.0.1 255.240.0.0 segment-routing mpls path name path1
mpls traffic-eng auto-tunnel p2p
mpls traffic-eng auto-tunnel p2p config unnumbered-interface loopback0
mpls traffic-eng auto-tunnel p2p tunnel-num min 10 max 100
```

## Verifying Segment Routing Auto-Tunnel Static Route

The command **show mpls traffic-eng service summary** displays all registered TE service clients and statistics that use TE auto tunnel.

```
Device# show mpls traffic-eng service summary

Service Clients Summary:
  Client: BGP TE
    Client ID                :0
    Total P2P tunnels        :1
    P2P add requests         :6
    P2P delete requests      :5
    P2P add falis            :0
    P2P delete falis         :0
    P2P notify falis         :0
    P2P notify succs         :12
    P2P replays              :0
  Client: ipv4static
    Client ID                :1
    Total P2P tunnels        :1
    P2P add requests         :6
    P2P delete requests      :5
    P2P add falis            :0
    P2P delete falis         :0
    P2P notify falis         :0
    P2P notify succs         :85
    P2P replays              :0
```

The command **show mpls traffic-eng auto-tunnel p2p** displays the peer-to-peer (P2P) auto tunnel configuration and operation status.

```
Device# show mpls traffic-eng auto-tunnel p2p

State: Enabled
  p2p auto-tunnels: 2 (up: 2, down: 0)
  Default Tunnel ID Range: 62336 - 64335
  Config:
   unnumbered-interface: Loopback0
   Tunnel ID range: 1000 - 2000
```

The command **show mpls traffic-eng tunnel summary** displays the status of P2P auto tunnel.

```
Device# show mpls traffic-eng tunnel summmary

Signalling Summary:
    LSP Tunnels Process:            running
    Passive LSP Listener:           running
    RSVP Process:                   running
    Forwarding:                     enabled
    auto-tunnel:
        p2p    Enabled  (1), id-range:1000-2000
    Periodic reoptimization:        every 3600 seconds, next in 1265 seconds
    Periodic FRR Promotion:         Not Running
    Periodic auto-bw collection:    every 300 seconds, next in 66 seconds
    SR tunnel max label push:       13 labels
    P2P:
      Head: 11 interfaces,   5234 active signalling attempts, 1 established
            5440 activations,  206 deactivations
            1821 failed activations
            0 SSO recovery attempts, 0 SSO recovered
      Midpoints: 0, Tails: 0
    P2MP:
      Head: 0 interfaces,   0 active signalling attempts, 0 established
            0 sub-LSP activations,  0 sub-LSP deactivations
            0 LSP successful activations,  0 LSP deactivations
            0 SSO recovery attempts, LSP recovered: 0 full, 0 partial, 0 fail
      Midpoints: 0, Tails: 0
```

```
Bidirectional Tunnel Summary:
    Tunnel Head: 0 total, 0 connected, 0 associated, 0 co-routed
    LSPs Head:   0 established, 0 proceeding, 0 associated, 0 standby
    LSPs Mid:    0 established, 0 proceeding, 0 associated, 0 standby
    LSPs Tail:   0 established, 0 proceeding, 0 associated, 0 standby

AutoTunnel P2P Summary:
    ipv4static:
        Tunnels: 1 created, 1 up, 0 down
    Total:
        Tunnels: 1 created, 1 up, 0 down
```

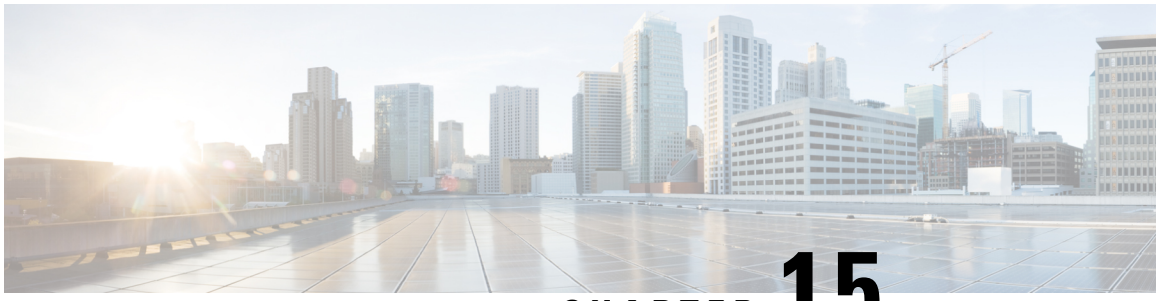
The command **show mpls traffic-eng tunnel auto-tunnel** only displays TE service auto tunnel.

```
Device# show mpls traffic-eng tunnel auto-tunnel detail

 P2P TUNNELS/LSPs:

Name: R1_t1000                             (Tunnel1000) Destination: 0.0.0.0 Ifhandle: 0x17
 (auto-tunnel for ipv4static)
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 1, (SEGMENT-ROUTING) type explicit (verbatim) path202 (Basis for Setup)

  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    Path Selection:
     Protection: any (default)
    Path-selection Tiebreaker:
      Global: not set   Tunnel Specific: not set   Effective: min-fill (default)
    Hop Limit: disabled [ignore: Verbatim Path Option]
    Cost Limit: disabled
    Path-invalidation timeout: 10000 msec (default), Action: Tear
    AutoRoute: disabled LockDown: disabled Loadshare: 0 [0] bw-based
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: explicit path option 1 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: enabled

  History:
    Tunnel:
      Time since created: 33 days, 20 hours, 29 minutes
      Time since path change: 10 days, 19 hours, 45 minutes
      Number of LSP IDs (Tun_Instances) used: 1646
    Current LSP: [ID: 1646]
      Uptime: 10 days, 19 hours, 45 minutes
    Prior LSP: [ID: 1645]
      ID: path option unknown
      Removal Trigger: signalling shutdown
  Tun_Instance: 1646
  Segment-Routing Path Info (IGP information is not used)
    Segment0[First Hop]: 0.0.0.0, Label: 16002
    Segment1[ - ]: Label: 16006
```

The command **show mpls traffic-eng tunnel brief** displays auto tunnel information.

```
Device# show mpls traffic-eng tunnel brief

Signalling Summary:
    LSP Tunnels Process:            running
```

```
    Passive LSP Listener:           running
    RSVP Process:                   running
    Forwarding:                     enabled
    auto-tunnel:
        p2p    Enabled  (2), id-range:1000-2000

    Periodic reoptimization:        every 3600 seconds, next in 406 seconds
    Periodic FRR Promotion:         Not Running
    Periodic auto-bw collection:    every 300 seconds, next in 107 seconds
    SR tunnel max label push:       13 labels

P2P TUNNELS/LSPs:
TUNNEL NAME                       DESTINATION      UP IF     DOWN IF   STATE/PROT
R1_t1                             66.66.66.66      -         -         up/down
R1_t2                             66.66.66.66      -         -         up/up
R1_t3                             66.66.66.66      -         -         up/up
R1_t10                            66.66.66.66      -         -         up/up
SBFD tunnel                       33.33.33.33      -         -         up/up
SBFD Session configured: 1        SBFD sessions UP: 1
```



CHAPTER 15

# Segment Routing MPLS OAM Support

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, IGP prefix SID FEC type, and partially IGP adjacency-SID FEC type for SR-TE functionality.


- Feature Information for Segment Routing OAM Support, on page 161
- Restrictions for Segment Routing OAM MPLS Support, on page 162
- Information About Segment Routing MPLS OAM Support, on page 162
- How to Diagnose Segment Routing with LSP Ping and Trace Route Nil FEC Target, on page 164
- Example for LSP Ping Nil FEC Target Support, on page 165
- Path Validation in Segment Routing Network, on page 166
- Configuring Segment Routing MPLS Traffic Engineering for MPLS Ping and Traceroute, on page 168
- Configuring Segment Routing MPLS IGP for MPLS Ping and Traceroute, on page 169
- Verifying Segment Routing OAM Using Cisco IOS CLI, on page 170


# Feature Information for Segment Routing OAM Support

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

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

***Table 14: Feature Information for Segment Routing OAM Support***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing OAM Support | Cisco IOS XE Amsterdam 17.3.2 | The Segment Routing OAM feature provides support for Nil-FEC (forwarding equivalence classes) LSP Ping and Traceroute functionality.<br><br>The Nil-FEC LSP ping and traceroute operation are simply extension of regular MPLS ping and trace route. |

| Feature Name | Releases | Feature Information |
|---|---|---|
| Verifying Segment Routing OAM Using CLI | Cisco IOS XE Amsterdam 17.3.2 | This feature provides the Command Line Interfaces (CLIs) that are needed to verify segment routing OAM feature(s). Ping and traceroute commands display the operation and output over IGP (OSPF SR, IS-IS SR), and SR-TE.<br>The following commands were introduced or modified:<br>**ping mpls ipv4 4.4.4.4/32 fec-type isis sr-path-type ip verbose**, **ping mpls ipv4 4.4.4.4/32 fec-type isis sr-path-type sid verbose**, **ping mpls ipv4 4.4.4.4/32 fec-type isis sr-path-type strict-sid verbose**, **ping mpls ipv4 4.4.4.4/32 fec-type ospf sr-path-type ip verbose**, **ping mpls ipv4 4.4.4.4/32 fec-type ospf sr-path-type sid verbose**, **traceroute mpls traffic-eng tunnel 1005 verbose**, **traceroute mpls ipv4 55.5.5.5/32 output interface tunnel1 force-explicit-null**, **traceroute mpls ipv4 4.4.4.4/32 fec-type ospf sr-path-type ip verbose**, **traceroute mpls ipv4 4.4.4.4/32 fec-type ospf sr-path-type sid verbose**, **traceroute mpls multipath ipv4 4.4.4.4/32 fec-type ospf sr-path-type sid verbose**. |

# Restrictions for Segment Routing OAM MPLS Support

- Ping and traceroute are unsupported with SR-TE static auto tunnel, BGP Dynamic TE, and on-demand next hop auto tunnels.
- Strict-SID option is not supported by the path installed by OSPF.
- MPLS traceroute does not support popping of two explicit null labels in one node.
- Rerouting the path to IP over MPLS segment without using Layer3 VPN is not supported due to IP routing destination not being a MPLS FEC.

# Information About Segment Routing MPLS OAM Support

## Segment Routing OAM Support

The Nil-FEC LSP ping and traceroute operations are extensions of regular MPLS ping and traceroute . Nil-FEC LSP Ping/Trace functionality support Segment Routing and MPLS Static. It also act as an additional diagnostic tool for all other LSP types. This feature allows operators to test any label stack to specify the following:

- label stack
- outgoing interface
- nexthop address

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

# Benefits of Segment Routing OAM Support

- The feature enables the MPLS OAM functionality in the Segment Routing Network where the traffic is engineering via SR-TE tunnels or native SR forwarding.
- In traditional MPLS networks, source node chooses the path based on hop by hop signaling protocols such as LDP or RSVP-TE. In Segment Routing Networks, the path is specified by set of segments which are advertised by the IGP protocols (currently OSPF and ISIS).
- As the volume of services offered using SR increase, it is important that the operator essentially is able to do the connectivity verification and the fault isolation in the SR architecture.
- The segment assignment is not based on hop by hop protocols as in traditional MPLS network, any broken transit node could lead to null routes, which could lead to undesired trraffic behavior.
- Both SR and SR-TE supports load balancing, it is important to trace all the ECMP paths available between source and target routers. The features offers the multipath traceroute support for both TE and native SR paths.
- The following are the main benefits of Segment Routing-OAM Support:
  - **Operations**: Network monitoring and fault management.
  - **Administration**: Network discovery and planning.
  - **Maintenance**: Corrective and preventive activities, minimize occurrences and impact of failures.

# Segment Routing MPLS Ping

MPLS ping and traceroute are extendable by design. You can add SR support by defining new FECs and/or additional verification procedures. MPLS ping verifies MPLS data path and performs the following:

- Encapsulates echo request packet in MPLS labels.
- Measures coarse round trip time.
- Measures coarse round trip delay.

# Segment Routing MPLS Traceroute

MPLS ping and traceroute are extendable by design. You can add SR support by defining new forwarding equivalence classes (FECs) and/or additional verification procedures. MPLS traceroute verifies forwarding and control plane at each hop of the LSP to isolate faults. Traceroute sends MPLS echo requests with monotonically increasing time-to-live (TTL), starting with TTL of 1. Upon TTL expiry, transit node processes the request in software and verifies if it has an LSP to the target FEC and intended transit node. The transit node sends echo reply containing return code specifying the result of above verification and label stack to reach the next-hop, as well as ID of the next-hop towards destination, if verification is successful. Originator

processes echo reply to build the next echo request containing TTL+1. Process is repeated until the destination replies that it is the egress for the FEC.

## LSP Ping Operation for Nil FEC target

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

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

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

## Using LSP Ping for Nil FEC Target

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

```
ping mpls nil-fec labels <comma separated labels> output interface <tx-interface> nexthop
<nexthop ip addr>
[repeat <count>]
[size <size> | sweep <min_size> <max_size> <increment>]
[timeout <seconds>]
[interval <milliseconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr>]
[exp <exp-value>]
[pad <pattern>]
[ttl <ttl>]
[reply [mode [ipv4 | router-alert | no-reply]]
[dscp <dscp-bits>]
[pad-tlv]]
[verbose]
[force-disposition ra-label]
{dsmap | ddmap [l2ecmp]} [hashkey {none | {ipv4 | ipv4-label-set {bitmap <bitmap_size>}}]
```

For more information, refer ping mpls.

## Using LSP Traceroute for Nil FEC Target

```
trace mpls nil-fec labels <comma separated labels> output interface <tx-interface>} nexthop
 <nexthop ip addr>
[timeout <seconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
```

```
[source <addr> ]
[exp <exp-value>]
[ttl <ttl-max>]
[reply [mode [ipv4 | router-alert | no-reply]]
[dscp <dscp-bits>]
[pad-tlv]]
[flags {fec | ttl}]
[hashkey ipv4 | ipv4-label-set {bitmap <bitmap_size>}]
```

For more information, refer to the traceroute mpls.

# Example for LSP Ping Nil FEC Target Support

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

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

```
16252      16252      9.9.9.7/32       0            Et1/1      90.1.1.5
16253      Pop Label  9.9.9.4/32       0            Et0/2      102.102.102.2
17000      17000      16.16.16.16/32   0            Et1/0      30.1.1.3
17002      17002      1.1.1.2/32       0            Et1/0      30.1.1.3
           17002      1.1.1.2/32       0            Et1/1      90.1.1.5

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

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

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


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

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

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

# Path Validation in Segment Routing Network

The MPLS OAM mechanisms help with fault detection and isolation for a MPLS data-plane path by the use of various target FEC stack sub-TLVs that are carried in MPLS echo request packets and used by the responder for FEC validation. While it is obvious that new sub-TLVs need to be assigned for segment routing, the unique nature of the segment routing architecture raises the need for additional operational considerations for path validation.

The forwarding semantic of Adjacency Segment ID is to pop the Segment ID and send the packet to a specific neighbor over a specific link. A malfunctioning node may forward packets using Adjacency Segment ID to an incorrect neighbor or over an incorrect link. The exposed Segment ID (of an incorrectly forwarded Adjacency Segment ID) might still allow such packet to reach the intended destination, although the intended strict traversal has been broken. MPLS traceroute may help with detecting such a deviation.

The format of the following Segment ID sub-TLVs follows the philosophy of Target FEC Stack TLV carrying FECs corresponding to each label in the label stack. This allows LSP ping/traceroute operations to function when Target FEC Stack TLV contains more FECs than received label stack at responder nodes. Three new sub-TLVs are defined for Target FEC Stack TLVs (Type 1), Reverse-Path Target FEC Stack TLV (Type 16) and Reply Path TLV (Type 21).

```
sub-Type    Value Field
--------  ---------------
   34      IPv4 IGP-Prefix Segment ID
   35      IPv6 IGP-Prefix Segment ID
   36      IGP-Adjacency Segment ID
```

# MPLS Ping and Traceroute for IGP Prefix-SID FEC Type

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

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.

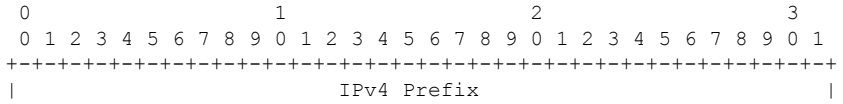
The MPLS LSP Traceroute feature is used to isolate the failure point of an LSP. It is used for hop-by-hop fault localization and path tracing. The MPLS LSP Traceroute feature relies on the expiration of the Time to Live (TTL) value of the packet that carries the echo request. When the MPLS echo request 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 IGPIPv4 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).

The network node which advertised the Node Segment ID is responsible for generating a FEC Stack Change sub-TLV with pop operation type for Node Segment ID, regardless of whether penultimate hop popping (PHP) is enabled or not.

The format is as below for IPv4 IGP-Prefix Segment ID:



```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Prefix Length  |    Protocol   |         Reserved              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The format is as below for IPv6 IGP-Prefix Segment ID:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                         IPv6 Prefix                           |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Prefix Length  |    Protocol   |              Reserved         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

# MPLS Ping and Traceroute for IGP-Adjacency Segment ID

The network node that is immediate downstream of the node which advertised the Adjacency Segment ID is responsible for generating FEC Stack Change sub-TLV for "POP" operation for Adjacency Segment ID.

The format is as below for IGP-adjacency SID:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Adj. Type   |    Protocol   |          Reserved             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                Local Interface ID (4 or 16 octets)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                Remote Interface ID (4 or 16 octets)           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                                                               ~
|          Advertising Node Identifier (4 or 6 octets)          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                                                               ~
|             Receiving Node Identifier (4 or 6 octets)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

# Configuring Segment Routing MPLS Traffic Engineering for MPLS Ping and Traceroute

```
ping mpls traffic-eng tunnel <tun-id>
[repeat <count>]
[size <size> | sweep <min_size> <max_size> <increment>]
[timeout <seconds>]
[interval <milliseconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr>]
[exp <exp-value>]
[pad <pattern>]
[ttl <ttl>]
[reply [mode [ipv4 | router-alert | no-reply]]
[dscp <dscp-bits>]
```

```
[pad-tlv]]
[verbose]
[output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
[{dsmap | ddmap [l2ecmp]} [hashkey {none | {ipv4 | ipv4-label-set {bitmap <bitmap_size>}}]

traceroute mpls [multipath] traffic-eng <tunnel-interface>
[timeout <seconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr> ]
[exp <exp-value>]
[ttl <ttl-max>]
[reply [mode [ipv4 | router-alert | no-reply]]
[pad-tlv]]
[output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
[flags {fec | ttl}]
[hashkey ipv4 | ipv4-label-set {bitmap <bitmap_size>}]
```

# Configuring Segment Routing MPLS IGP for MPLS Ping and Traceroute

```
ping mpls ipv4 <prefix/prefix_length> [fec-type [ldp | bgp | generic | isis | ospf]]
[sr-path-type [ip | sid | strict-sid]]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr>]
[exp <exp-value>]
[pad <pattern>]
[ttl <ttl>]
[reply [mode [ipv4 | router-alert | no-reply]]
[dscp <dscp-bits>]
[pad-tlv]]
[verbose]
[output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
[{dsmap | ddmap [l2ecmp]} [hashkey {none | {ipv4 | ipv4-label-set {bitmap <bitmap_size>}}]

traceroute mpls [multipath] ipv4 <prefix/prefix_length> [fec-type [ldp | bgp | generic |
isis | ospf]] [sr-path-type [ip | sid | strict-sid]]
[timeout <seconds>]
[destination <addr_start> [<addr_end> [<addr_incr_mask> | <addr_incr>]]]
[source <addr> ]
[exp <exp-value>]
[ttl <ttl-max>]
[reply [mode [ipv4 | router-alert | no-reply]]
[pad-tlv]]
[output {interface <tx-interface>} [nexthop <nexthop ip addr>]]
[flags {fec | ttl}]
[hashkey ipv4 | ipv4-label-set {bitmap <bitmap_size>}]
```

- fec-type: IPv4 Target FEC type, use head end auto detected FEC type by default.
- sr-path-type: Segment routing path type selection algorithm. Use IP imposition path, when option is specified.

# Verifying Segment Routing OAM Using Cisco IOS CLI

This section provides a summary on the main Command Line Interfaces (CLIs) that are needed to verify segment routing OAM feature(s). Ping and traceroute commands illustrate the operation and output over IGP (OSPF SR), ISIS SR, and SR-TE. Change the actual tunnel numbers and IP addresses based on the actual values needed and enabled in the configurations.

## Verifying Segment Routing Traffic Engineering OAM Operations

The following **traceroute** command displays SR-TE OAM operations:

```
SR_Device#traceroute mpls traffic-eng tunnel 1005 verbose
Tracing MPLS TE Label Switched Path on Tunnel1005 Active LSP, timeout is 2 seconds

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

Type escape sequence to abort.
  0 100.103.1.1 100.103.1.2 MRU 1500 [Labels: implicit-null/22/22 Exp: 0/0/0], RSC 0
l 1 100.103.1.2 103.104.1.2 MRU 1500 [Labels: implicit-null/22 Exp: 0/0] 3 ms, ret code 15,
 RSC 0
l 2 103.104.1.2 104.105.1.2 MRU 1500 [Labels: implicit-null Exp: 0] 3 ms, ret code 15, RSC
 0
! 3 104.105.1.2 2 ms, ret code 3



SR_Device#traceroute mpls ipv4 55.5.5.5/32 output interface tunnel1 force-explicit-null
Tracing MPLS Label Switched Path to 55.5.5.5/32, timeout is 2 seconds

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

Type escape sequence to abort.
  0 100.101.1.1 MRU 1500 [Labels: 26000/explicit-null Exp: 0/0]
L 1 100.101.1.2 MRU 1500 [Labels: 26000/explicit-null Exp: 0/0] 3 ms
L 2 101.104.1.2 MRU 1500 [Labels: implicit-null/explicit-null Exp: 0/0] 6 ms
! 3 104.105.1.2 3 ms
```

The following **tree traceroute** command displays SR-TE OAM operations in ECMP scenarios:

```
SR_Device#traceroute mpls multi traffic-eng tunnel 1 verbose
Starting LSP Multipath Traceroute for Tunnel1
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
'L' - labeled output interface, 'B' - unlabeled output interface,
'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
```

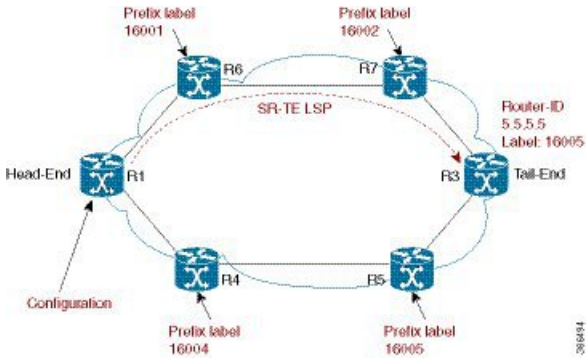
```
'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
'P' - no rx intf label prot, 'p' - premature termination of LSP,
'R' - transit router, 'I' - unknown upstream index,
'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
LL!
Path 0 found,
output interface Gi2 nexthop 100.101.1.2
source 50.0.0.0 destination 127.0.0.0
0 100.101.1.1 100.101.1.2 MRU 1500 [Labels: 26000 Exp: 0] multipaths 0
L 1 100.101.1.2 101.102.1.2 MRU 1500 [Labels: 26000 Exp: 0] ret code 8, RSC 0 multipaths 2
L 2 101.102.1.2 102.105.1.2 MRU 1500 [Labels: implicit-null Exp: 0] ret code 8, RSC 0
multipaths 1
! 3 102.105.1.2, ret code 3 multipaths 0
L!
Path 1 found,
output interface Gi2 nexthop 100.101.1.2
source 50.0.0.0 destination 127.0.0.1
0 100.101.1.1 100.101.1.2 MRU 1500 [Labels: 26000 Exp: 0] multipaths 0
L 1 100.101.1.2 101.104.1.2 MRU 1500 [Labels: 26000 Exp: 0] ret code 8, RSC 0 multipaths 2
L 2 101.104.1.2 104.105.1.2 MRU 1500 [Labels: implicit-null Exp: 0] ret code 8, RSC 0
multipaths 1
! 3 104.105.1.2, ret code 3 multipaths 0
LL!
Path 2 found,
output interface Gi3 nexthop 100.103.1.2
source 50.0.0.0 destination 127.0.0.0
0 100.103.1.1 100.103.1.2 MRU 1500 [Labels: 26000 Exp: 0] multipaths 0
L 1 100.103.1.2 102.103.1.1 MRU 1500 [Labels: 26000 Exp: 0] ret code 8, RSC 0 multipaths 2
L 2 102.103.1.1 102.105.1.2 MRU 1500 [Labels: implicit-null Exp: 0] ret code 8, RSC 0
multipaths 1
! 3 102.105.1.2, ret code 3 multipaths 0
L!
Path 3 found,
output interface Gi3 nexthop 100.103.1.2
source 50.0.0.0 destination 127.0.0.1
0 100.103.1.1 100.103.1.2 MRU 1500 [Labels: 26000 Exp: 0] multipaths 0
L 1 100.103.1.2 103.104.1.2 MRU 1500 [Labels: 26000 Exp: 0] ret code 8, RSC 0 multipaths 2
L 2 103.104.1.2 104.105.1.2 MRU 1500 [Labels: implicit-null Exp: 0] ret code 8, RSC 0
multipaths 1
! 3 104.105.1.2, ret code 3 multipaths 0
Paths (found/broken/unexplored) (4/0/0)
Echo Request (sent/fail) (10/0)
Echo Reply (received/timeout)
```

# Verifying Segment Routing OAM OSPF Using CLI

You need to specify the fec type specifically when performing the ping or traceroute for the prefixes which span across IGP boundaries. For example, when a prefix is redistributed to OSPF from ISIS domain then specify the fec type ISIS. When the ping or traceroute is performed within the IGP domain then you do not need to mention fec type explicitly. Provide generic fec type generic when the user does not know the IGP protocol on the destination node. When SR path type is not mentioned, default SR path type IP is taken.

The following topology is an example of a SR path type:

***Figure 19:***




The following ping commands are used to illustrate SR OAM when the underlying network is OSPF.

As per the above topology example, at the head end R1, SR-TE tunnel is created with the destination as R3. The SR-TE tunnel is created with explicit path option to pass through R6 and R7. The SR-TE path is, R1---R6----R7----R3, when the IP traffic ingress at R1.

```
Device#ping mpls ipv4 4.4.4.4/32 fec-type ospf sr-path-type ip verbose
Sending 2, 72-byte MPLS Echos to IGP Prefix SID(OSPF) FEC 5.5.5.5/32,
     timeout is 2 seconds, send interval is 0 msec:
Select segment routing IP imposition path.

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

Type escape sequence to abort.
!    size 72, reply addr 2.4.0.4, return code 3
!    size 72, reply addr 2.4.0.4, return code 3

Success rate is 100 percent (2/2), round-trip min/avg/max = 1/1/1 ms
 Total Time Elapsed 4 ms
```

In the same topology, when the incoming traffic is labeled traffic, then the following two ECMP paths are chosen for the forwarding:

- R1---R6----R7----R3
- R1---R4----R5----R3

**Note** Using the multipath option, both the paths can be traced for the destination.

```
Device# ping mpls ipv4 4.4.4.4/32 fec-type ospf sr-path-type sid verbose
Sending 1, 72-byte MPLS Echos to IGP Prefix SID(OSPF) FEC 5.5.5.5/32,
     timeout is 2 seconds, send interval is 0 msec:
Select segment routing prefix SID path.
```

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

Type escape sequence to abort.
!    size 72, reply addr 2.4.0.4, return code 3

Success rate is 100 percent (1/1), round-trip min/avg/max = 1/1/1 ms
 Total Time Elapsed 3 ms
```

The following **traceroute** commands display SR OAM when the underlying network is OSPF.

To trace the IP route path when the incoming traffic to R1 is the native IP, the below command is used at the end of R1.

```
Device#traceroute mpls ipv4 4.4.4.4/32 fec-type ospf sr-path-type ip verbose
Tracing MPLS Label Switched Path to IGP Prefix SID(OSPF) FEC 4.4.4.4/32, timeout is 2 seconds
Select segment routing IP imposition path.

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

Type escape sequence to abort.
 0 1.2.0.1 1.2.0.2 MRU 1500 [Labels: 16002/16005 Exp: 0/0], RSC 0

L 1 1.2.0.2 3.3.3.3 MRU 1500 [Labels: 16005 Exp: 0] 2 ms, ret code 8, RSC 0
L 2 3.3.3.3 3.4.0.4 MRU 1500 [Labels: implicit-null Exp: 0] 1 ms, ret code 8, RSC 0
! 3 3.4.0.4 1 ms, ret code 3


Device#traceroute mpls ipv4 4.4.4.4/32 fec-type ospf sr-path-type sid  verbose


Device#traceroute mpls multipath ipv4 4.4.4.4/32  fec-type ospf sr-path-type ip verbose
Type escape sequence to abort.
LL!
Path 0 found,
 output interface Et0/1 nexthop 1.2.0.2                              //path R1-R6-R7-R3
 source 1.1.1.1 destination 127.0.0.0
  0 1.2.0.1 1.2.0.2 MRU 1500 [Labels: 16666 Exp: 0] multipaths 0
L 1 1.2.0.2 2.4.0.4 MRU 1500 [Labels: 16666 Exp: 0] ret code 8, RSC 0 multipaths 1
L 2 2.4.0.4 4.6.0.6 MRU 1500 [Labels: implicit-null Exp: 0] ret code 8, RSC 0 multipaths 1
! 3 4.6.0.6, ret code 3 multipaths 0
LL!
Path 1 found,
 output interface Et0/2 nexthop 1.3.0.3                                //path R1-R4-R5-R3
 source 1.1.1.1 destination 127.0.0.0
  0 1.3.0.1 1.3.0.3 MRU 1500 [Labels: 16666 Exp: 0] multipaths 0
L 1 1.3.0.3 3.4.0.4 MRU 1500 [Labels: 16666 Exp: 0] ret code 8, RSC 0 multipaths 1
L 2 3.4.0.4 4.6.0.6 MRU 1500 [Labels: implicit-null Exp: 0] ret code 8, RSC 0 multipaths 1
! 3 4.6.0.6, ret code 3 multipaths 0

Paths (found/broken/unexplored) (2/0/0)
```

```
 Echo Request (sent/fail) (6/0)
 Echo Reply (received/timeout) (6/0)
 Total Time Elapsed 23 ms

Device#traceroute mpls multipath ipv4 4.4.4.4/32 fec-type ospf sr-path-type sid verbose
```

# Verifying Segment Routing OAM IS-IS Using CLI

The following **ping** commands are used to display SR OAM when the underlying network is IS-IS:

```
Device# ping mpls ipv4 4.4.4.4/32 fec-type isis sr-path-type ip verbose

Device# ping mpls ipv4 4.4.4.4/32 fec-type isis sr-path-type sid verbose

Device# ping mpls ipv4 4.4.4.4/32 fec-type isis sr-path-type strict-sid verbose

Device# ping mpls ipv4 4.4.4.4/32 sr-path-type ip verbose

Device# ping mpls ipv4 4.4.4.4/32 sr-path-type sid verbose

Device# ping mpls ipv4 4.4.4.4/32 sr-path-type strict-sid verbose
```

The following **traceroute** commands display SR OAM when the underlying network is IS-IS. When multipath option is enabled, all ECMP paths are returned.

```
Device# traceroute mpls ipv4 4.4.4.4/32 fec-type isis sr-path-type ip verbose

Device# traceroute mpls ipv4 4.4.4.4/32 fec-type isis sr-path-type sid  verbose

Device# traceroute mpls ipv4 4.4.4.4/32 fec-type isis sr-path-type strict-sid verbose

Device# traceroute mpls multipath ipv4 4.4.4.4/32  fec-type isis sr-path-type ip verbose

Device# traceroute mpls multipath ipv4 4.4.4.4/32 fec-type isis sr-path-type sid verbose

Device# traceroute mpls multipath ipv4 4.4.4.4/32 fec-type isis sr-path-type strict-sid
verbose
```

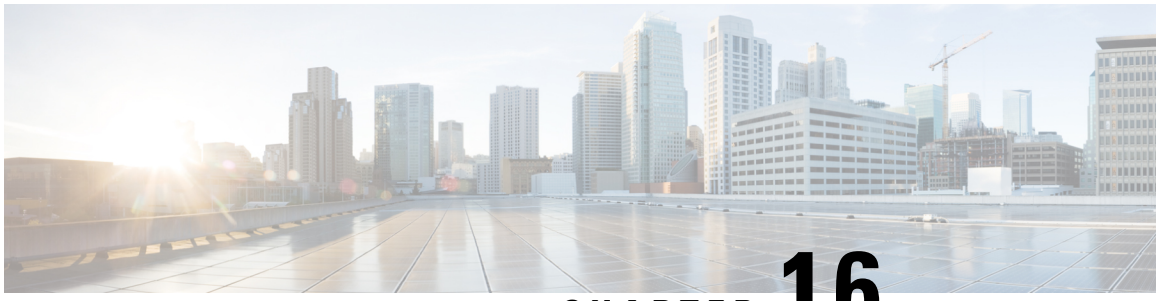
# Verifying MPLS Ping and Traceroute for IGP Segment ID

Use the following command SR network with IGP validation:

```
ping|traceroute mpls [multipath] ipv4 <prefix> [fec-type bgp |generic|ldp|isis|ospf]
[sr-path-type ip|sid|strict-sid]
```

Use the following command to verify MPLS TE tunnel OAM when the tunnel LSP is a SR-TE LSP.

```
ping|traceroute mpls traffic-eng tunnel <tunnelid>
```



CHAPTER 16

# Using Seamless BFD with Segment Routing

The Segment Routing TE feature provides information support for Seamless Bidirectional Forwarding Detection (S-BFD).


- Feature Information for Seamless BFD with Segment Routing, on page 175
- Restrictions For Using Seamless BFD with Segment Routing, on page 176
- Information About Seamless BFD with Segment Routing, on page 176
- How to Configure Seamless BFD with Segment Routing, on page 177
- Additional References for Seamless BFD with Segment Routing, on page 179


## Feature Information for Seamless BFD with Segment Routing

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

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

***Table 15: Feature Information for Segment Routing TE Feature***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing TE Feature | Cisco IOS XE Amsterdam 17.3.2 | Seamless Bidirectional Forwarding Detection (S-BFD), is a simplified mechanism for using BFD with a large proportion of negotiation aspects eliminated, thus providing benefits such as quick provisioning, as well as improved control and flexibility for network nodes initiating path monitoring.<br><br>The following commands were introduced or modified: **address-family ipv4 strict-spf**, **bfd-template single-hop**, **index range**, **sbfd local-discriminator**, **show bfd neighbor**, **show isis segment-routing**, **show mpls forwarding-table**, **show mpls traffic tunnel**, **show mpls traffic-engineering**. |

# Restrictions For Using Seamless BFD with Segment Routing

### Restrictions for Seamless-Birdirectional Forwarding (S-BFD)

- Seamless-Birdirectional Forwarding (S-BFD) supporting IPv4 only for segment routing traffic engineering (SR-TE). IPv6 is not supported.
- Single hop S-BFD session is only supported.
- RSVP-TE does not support S-BFD.

# Information About Seamless BFD with Segment Routing

## Bidirectional Forwarding Detection and Seamless-Bidirectional Forwarding Detection (S-BFD)

Bidirectional Forwarding Detection (BFD) is a detection protocol designed to provide fast forwarding path failure detection times for all media types, encapsulations, topologies, and routing protocols.

BFD provides a consistent failure detection method for network administrators, in addition to fast forwarding path failure detection. Because the network administrator can use BFD to detect forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol hello mechanisms, network profiling and planning will be easier, and reconvergence time will be consistent and predictable.

Seamless Bidirectional Forwarding Detection (S-BFD), is a simplified mechanism for using BFD with a large proportion of negotiation aspects eliminated, thus providing benefits such as quick provisioning, as well as improved control and flexibility for network nodes initiating path monitoring.

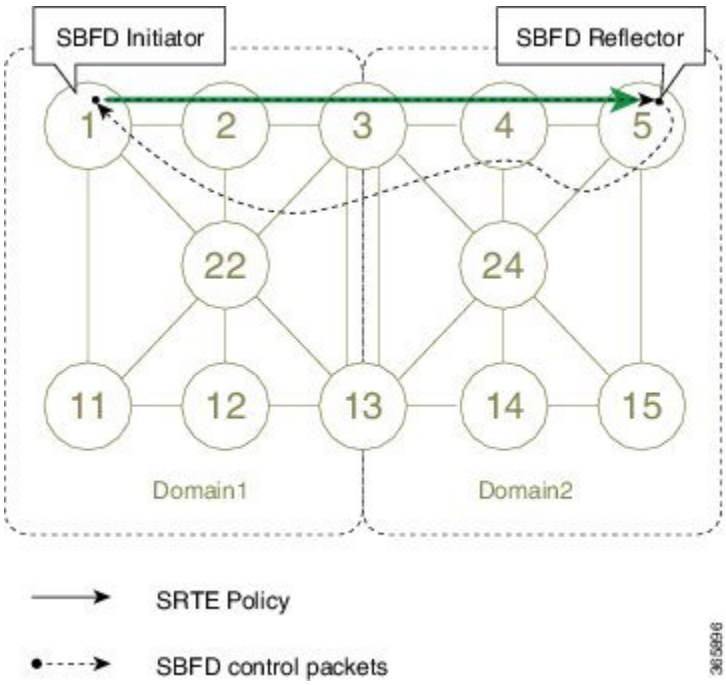
If SBFD session fails, S-BFD brings down the SR-TE session. S-BFD also provides faster session bring up due to less control packets exchange. S-BFD is associated with SR-TE to bring a session up quickly. The BFD state is only maintained at head end thereby reducing overhead.

S-BFD implements support for RFC 7880, RFC 7881 on segment routing.

## Initiators and Reflectors

SBFD runs in an asymmetric behavior, using initiators and reflectors. The following figure illustrates the roles of an SBFD initiator and reflector.

*Figure 20: SBFD Initiator and Reflector*




The initiator is an SBFD session on a network node that performs a continuity test to a remote entity by sending SBFD packets. The initiator injects the SBFD packets into the segment-routing traffic-engineering (SRTE) policy. The initiator triggers the SBFD session and maintains the BFD state and client context.

The reflector is an SBFD session on a network node that listens for incoming SBFD control packets to local entities and generates response SBFD control packets. The reflector is stateless and only reflects the SBFD packets back to the initiator.

A node can be both an initiator and a reflector, thereby allowing you to configure different SBFD sessions.

S-BFD can be enabled and supported for SR-TE IPv4, but IPv6 is not supported. For SR-TE, S-BFD control packets are label switched in forward and reverse direction. For S-BFD, the tail end is the reflector node. Other nodes cannot be a reflector. When using S-BFD with SR-TE, if the forward and return directions are label switched paths, S-BFD need not be configured on the reflector node.

# How to Configure Seamless BFD with Segment Routing

## Configuring Seamless-Bidirectional Forwarding Detection (S-BFD) for Segment Routing

S-BFD must be enabled on both initiator and reflector nodes.

**Note** When using S-BFD with SR-TE, if the forward and return directions are label switched paths, S-BFD need not be configured on the reflector node.

## Enabling Seamless Bidirectional Forwarding Detection (S-BFD) on the Reflector Node

Perform this task to configure S-BFD on the reflector node.

```
sbfd local-discriminator 55.55.55.55
```

## Enabling Seamless Bidirectional Forwarding Detection (S-BFD) on the Initiator Node

Perform this task to configure S-BFD on the initiator node.

```
bfd-template single-hop ABC
 interval min-tx 300 min-rx 300 multiplier 10
```

## Enabling Segment Routing Traffic Engineering Tunnel with Seamless-Bidirectional Forwarding (S-BFD)

```
interface Tunnel56
 ip unnumbered Loopback11
 tunnel mode mpls traffic-eng
 tunnel destination 55.55.55.55 */IP address of Reflector node/*
 tunnel mpls traffic-eng path-option 1 dynamic segment-routing
 tunnel mpls traffic-eng bfd sbfd ABC
!
end
```

## Verifying S-BFD Configuration

### SUMMARY STEPS

1. **show mpls traffic-engineering tunnel** *tunnel-name*
2. **show bfd neighbors**

### DETAILED STEPS

**Step 1** **show mpls traffic-engineering tunnel** *tunnel-name*

Verifies the SR TE state and the S-BFD session state.

**Example:**

```
Router# sh mpls traffic-eng tunnel tunnel 56

    Name: R1_t56                             (Tunnel56) Destination: 55.55.55.55
      Status:
        Admin: up         Oper: up     Path: valid       Signalling: connected
        path option 1, (SEGMENT-ROUTING) type dynamic (Basis for Setup, path weight 12)

      Config Parameters:
        Bandwidth: 0        kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF
        Metric Type: TE (default)
        Path Selection:
         Protection: any (default)
        Path-selection Tiebreaker:
          Global: not set   Tunnel Specific: not set   Effective: min-fill (default)
        Hop Limit: disabled
        Cost Limit: disabled
        Path-invalidation timeout: 10000 msec (default), Action: Tear
        AutoRoute: disabled LockDown: disabled Loadshare: 0 [0] bw-based
        auto-bw: disabled
```

```
  Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No

SBFD configured with template: ABC
  Session type: CURRENT        State: UP       SBFD handle: 0x3
  LSP ID: 1
  Last uptime duration: 3 minutes, 35 seconds
  Last downtime duration: --
    Active Path Option Parameters:
  State: dynamic path option 1 is active
  BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled
Node Hop Count: 2
    History:
  Tunnel:
    Time since created: 4 minutes, 3 seconds
    Number of LSP IDs (Tun_Instances) used: 1
  Current LSP: [ID: 1]
    Uptime: 3 minutes, 36 seconds
Tun_Instance: 1
Segment-Routing Path Info (isis  level-2)
  Segment0[Link]: 12.12.12.1 - 12.12.12.2, Label: 48
  Segment1[Link]: 25.25.25.2 - 25.25.25.5, Label: 35 !
```

**Step 2** **show bfd neighbors**

Verifies that BFD neighbors are established properly.

**Example:**

```
Router# show bfd neighbors

    MPLS-TE SR Sessions
    Interface      LSP ID(Type)                  LD/RD         RH/RS     State
    Tunnel56          1 (SR)                  4097/926365495  Up        Up
```

# Additional References for Seamless BFD with Segment Routing

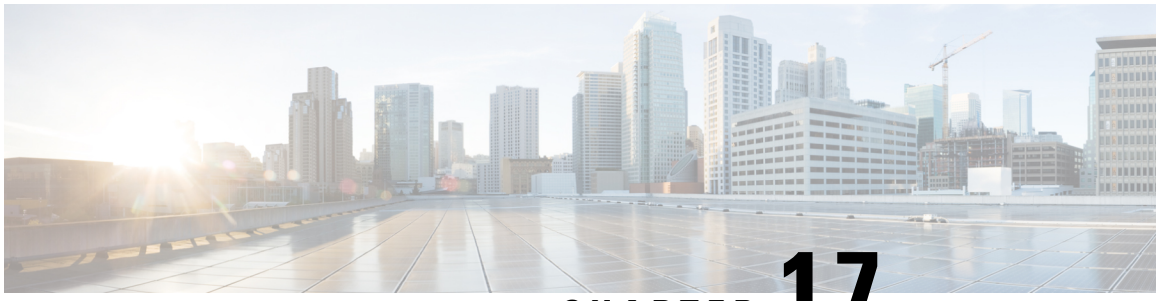
**Related Documents**

| Related Topic | Document Title |
|---|---|
| Segment Routing Traffic Engineering configuration | *Segment Routing -Traffic Engineering* |

***Table 16: Standards and RFC***

| Standard/RFC | Title |
|---|---|
| draft-akiya-bfd-seamless-base-03 | Seamless Bidirectional Forwarding Detection (S-BFD) |
| draft-ietf-isis-segment-routing-extensions-07 | IS-IS Extensions for Segment Routing |
| draft-ietf-spring-segment-routing-09 | Segment Routing Architecture |
| RFC 7880 | Seamless Bidirectional Forwarding Detection (S-BFD) |

| Standard/RFC | Title |
|---|---|
| RFC 7881 | Seamless Bidirectional Forwarding Detection (S-BFD) for IPv4, IPv6, and MPLS |



CHAPTER 17

# Using SSPF with Segment Routing

The Segment Routing TE feature provides information support for the Strict Shortest Path First (SPF).


- Feature Information for SSPF with Segment Routing, on page 181
- Information About SSPF with Segment Routing, on page 181
- How to Configure SSPF with Segment Routing, on page 182
- Additional References for SSPF with Segment Routing, on page 184


# Feature Information for SSPF with Segment Routing

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

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

***Table 17: Feature Information for Segment Routing SSPF Feature***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing TE Feature | Cisco IOS XE Amsterdam 17.3.2 | The Segment Routing TE feature provides information support for the Strict Shortest Path First (SPF)..<br><br>The following commands were introduced or modified: **address-family ipv4 strict-spf**, **bfd-template single-hop**, **index range**, **sbfd local-discriminator**, **show bfd neighbor**, **show isis segment-routing**, **show mpls forwarding-table**, **show mpls traffic tunnel**, **show mpls traffic-engineering**. |

# Information About SSPF with Segment Routing

## Strict Shortest Path First

Segment Routing supports the following two algorithms:

- Algorithm 0: This is a Shortest Path First (SPF) algorithm based on link metric. This shortest path algorithm is computed by the Interior gateway protocol (IGP).
- Algorithm 1: This is a Strict Shortest Path First (SSPF) algorithm based on link metric. The algorithm 1 is identical to algorithm 0 but requires that all nodes along the path honor the SPF routing decision. Local policy does not alter the forwarding decision. For example, a packet is not forwarded through locally engineered path.

Different SIDs are associated with the same prefix for each algorithm.

Strict Shortest Path First is supported by default - but strict SIDs must be configured for at least one node address on each node supporting Segment Routing.

# Approaches for Configure Strict Shortest Path First

The two approaches to configure Strict SFP are as follows:

- Using the **connect-prefix-sid-map** command—Strict SFP is configured globally on all the nodes. For a network to be Strict SFP-aware (that is, for ISIS to populate Strict SPF), all nodes must be configured with a local Strict SFP SID.
- Using Segment-routing Mapping Server—One node in the network is configured as mapping server and the remaining nodes act as a client.

# How to Configure SSPF with Segment Routing

## Configuring Strict Shortest Path First (SPF)

### Enabling Strict Shortest Path First Using the connect-prefix-sid-map command

#### Enabling Shortest Path First on a Provider-Edge Device

When enabling Strict Shortest Path First using the **connect-prefix-sid-map** command, the Strict Shortest Path First (SPF) must be configured on the provider-edge device first and then on the node devices. The following is a sample configuration code snippet to enable Strict Shortest Path First on a provider-edge device.

```
segment-routing mpls
connected-prefix-sid-map
  address-family ipv4
   10.10.10.10/32 index 100 range 1
  exit-address-family
  address-family ipv4 strict-spf
   10.10.10.10/32 index 1000 range 1 ------------------configure strict SPF locally
  exit-address-family
```

#### Enabling Shortest Path First on a Node Device

The following is a sample configuration code snippet to enable Strict Shortest Path First on a node in the network and must be enabled on all nodes in a network.

```
segment-routing mpls
```

```
connected-prefix-sid-map
  address-family ipv4
   20.20.20.20/32 index 110 range 1
  exit-address-family
  address-family ipv4 strict-spf
   20.20.20.20/32 index 1100 range 1
  exit-address-family
```

# Enabling Strict Shortest Path First Using Segment Routing Mapping Server

## Configuring a Node as Segment Routing Mapping Server

The following is a sample configuration code snippet to configure a node as Segment Routing Mapping Server.

```
segment-routing mpls
mapping-server
  prefix-sid-map
   address-family ipv4
    10.10.10.10/32 index 100 range 1
    20.20.20.20/32 index 110 range 1
    30.30.30.30/32 index 120 range 1
    40.40.40.40/32 index 130 range 1
    50.50.50.50/32 index 140 range 1
   exit-address-family
   address-family ipv4 strict-spf
    10.10.10.10/32 index 1000 range 1
    20.20.20.20/32 index 1100 range 1
    30.30.30.30/32 index 1200 range 1
    40.40.40.40/32 index 1300 range 1
    50.50.50.50/32 index 1400 range 1
    100.100.100.100/32 index 2000 range 1
   exit-address-family
```

## Configuring the Segment Routing Mapping Server to Advertise and Receive Local Prefixes

The following is a sample configuration code snippet to configure a Segment Routing Mapping Server to advertise and receive local prefixes.

```
router isis SR
segment-routing mpls
 segment-routing prefix-sid-map advertise-local
 segment-routing prefix-sid-map receive
```

## Verifying ISIS Advertises the SIDs

The following is a sample configuration code snippet to verify that ISIS advertises the SIDs.

```
Router# show isis segment-routing prefix-sid-map advertise strict-spf
Tag SR:
IS-IS Level-1 advertise prefix-sid maps:
Prefix               SID Index    Range        Flags
10.10.10.10/32       1000         1
20.20.20.20/32       1100         1
30.30.30.30/32       1200         1
40.40.40.40/32       1300         1
50.50.50.50/32       1400         1
100.100.100.100/32   2000         1
Tag SR:
IS-IS Level-2 advertise prefix-sid maps:
Prefix               SID Index    Range        Flags
10.10.10.10/32       1000         1
20.20.20.20/32       1100         1
30.30.30.30/32       1200         1
```

```
40.40.40.40/32       1300         1
50.50.50.50/32       1400         1
100.100.100.100/32   2000         1
```

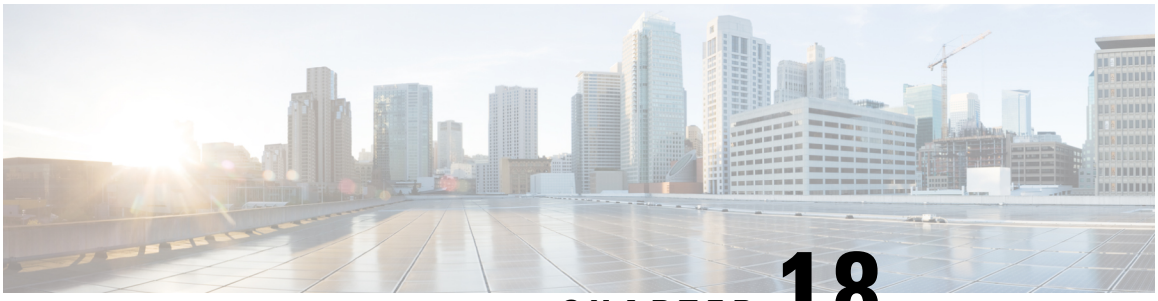
The following is a sample configuration code snippet to verify that a provider-edge device receives Strict Shortest Path First SID from SRMS Server.

```
Router# show isis segment-routing prefix-sid-map receive strict-spf

Tag SR:
IS-IS Level-1 receive prefix-sid maps:
Host                  Prefix              SID Index    Range        Flags
P1                    10.10.10.10/32      1000         1
                      20.20.20.20/32      1100         1
                      30.30.30.30/32      1200         1
                      40.40.40.40/32      1300         1
                      50.50.50.50/32      1400         1
                      100.100.100.100/32  2000         1
Tag SR:
IS-IS Level-2 receive prefix-sid maps:
Host                  Prefix              SID Index    Range        Flags
P1                    10.10.10.10/32      1000         1
                      20.20.20.20/32      1100         1
                      30.30.30.30/32      1200         1
                      40.40.40.40/32      1300         1
                      50.50.50.50/32      1400         1
                      100.100.100.100/32  2000         1
```

# Additional References for SSPF with Segment Routing

### Related Documents

| Related Topic | Document Title |
|---|---|
| Segment Routing Traffic Engineering configuration | *Segment Routing -Traffic Engineering* |



CHAPTER 18

# Dynamic PCC

The Stateful Path Computation Element Protocol(PCEP) enables a router to report and optionally delegate Label Switched Paths (LSPs) which is established using either Resource Reservation Protocol (RSVP) protocol or Segment Routing Traffic Engineering (SR-TE) to a stateful Path Computation Element (PCE).

An LSP delegated to a PCE can be updated by the PCE and a stateful PCE can compute and provide the path of an LSP to the Path Computation Client (PCC).

SR-TE and RSVP-TE LSPs require link-state routing protocols such as OSPF or ISIS to distribute and learn traffic engineering topology. A stateful PCE can learn the traffic engineering topology through BGP Link-State protocol. You can use the verbatim path option in the case when all or some of the intermediate nodes in a network do not support IGP extensions for TE.


- Information About Dynamic PCC, on page 185
- How to Configure Dynamic PCC, on page 186
- Verifying Dynamic PCC, on page 187
- Verifying Verbatim Path Option With Dynamic PCC, on page 190
- Feature Information for Dynamic PCC, on page 191


# Information About Dynamic PCC

## Path Computation Element Protocol Functions

A Path Computation Element Protocol (PCEP) session is a TCP session between a PCC and a PCE with protocol messages. The PCEP functions are verified based on the PCC functions. The configuration and verification show that the request is accepted and path computation is provided based on PCReq message from the client. The passive reporting enables a router to report a tunnel instead of delegating it to a PCE. The PCE is aware of the tunnel even though it cannot modify the tunnel.

PCEP functions are useful when a network has both router-controlled and PCE delegated tunnels. The PCE is aware of both the tunnels and can make an accurate decision on path computation.

## Redundant Path Computation Elements

For redundancy it may be required to deploy redundant PCE servers. A PCC uses precedence to select stateful PCEs for delegating LSPs. Precedence can take any value between 0 and 255. The default precedence value is 255. When there are multiple stateful PCEs with active PCEP session, PCC chooses the PCE with the lowest

precedence value. In case where primary PCE server session goes down, PCC router re-delegates all tunnels to next available PCE server. You can use the following CLIs in the case of redundant PCEs:

```
R2(config)#mpls traffic-eng pcc peer 77.77.77.77 source 22.22.22.22 precedence 255
R2(config)#mpls traffic-eng pcc peer 88.88.88.88  source 22.22.22.22 precedence 100
!
end
```

In the above example PCE server with IP address 88.88.88.88 is the primary PCE server since it has lower precedence value.

# How to Configure Dynamic PCC

## Configuring Dynamic PCC Globally

Perform the following task to configure dynamic PCC globally

```
enable
configure terminal
mpls traffic-eng tunnels
mpls traffic-eng pcc peer 10.0.0.1    ----□(10.0.0.1 is the PCE server address)
mpls traffic-eng pcc report-all
end
```

**Note** **mpls traffic-eng pcc report-all** is not mandatory for PCE/PCC basic operational delegated tunnels. It is required to report locally calculated LSPs to the PCE server.

## Configuring Dynamic PCC on an Interface

Perform the following task to configure dynamic PCC on an interface

```
interface Tunnel1
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 7.7.7.7
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 5 5
 tunnel mpls traffic-eng bandwidth 200
 tunnel mpls traffic-eng path-option 10 dynamic pce segment-routing
end
```

## Configuring Dynamic PCC With Verbatim Path Option

To enable Dynamic PCC with verbatim path option, use the following CLI under the SR-TE tunnel interface:

```
R1#
interface Tunnel2
ip unnumbered Loopback11
```

```
tunnel mode mpls traffic-eng
tunnel destination 66.66.66.66
tunnel mpls traffic-eng autoroute destination
tunnel mpls traffic-eng path-option 1 dynamic segment-routing pce verbatim
```

# Verifying Dynamic PCC

The following sample output is from the **show pce client peer detail** command.

```
Device# show pce client peer detail

PCC's peer database:
--------------------

Peer address: 1.1.1.1
  State up
  Capabilities: Stateful, Update, Segment-Routing
  PCEP has been up for: 23:44:58
  PCEP session ID: local 1, remote: 0
  Sending KA every 30 seconds
  Minimum acceptable KA interval: 20 seconds
  Peer timeout after 120 seconds
  Statistics:
    Keepalive messages: rx    2798 tx    2112
    Request messages:   rx       0 tx      32
    Reply messages:     rx      32 tx       0
    Error messages:     rx       0 tx       0
    Open messages:      rx       1 tx       1
    Report messages:    rx       0 tx      57
    Update messages:    rx      72 tx       0
```

The following sample output is from the **show mpls traffic-eng tunnels tunnel 1** command which shows the LSP details.

```
Device# show mpls traffic-eng tunnels tunnel 1

Name: d1_t1                           (Tunnel1) Destination: 7.7.7.7
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 10, (SEGMENT-ROUTING) (PCE) type dynamic (Basis for Setup, path weight 0)

  Config Parameters:
    Bandwidth: 200      kbps (Global)  Priority: 5  5   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    Path Selection:
     Protection: any (default)
    Path-selection Tiebreaker:
      Global: not set   Tunnel Specific: not set   Effective: min-fill (default)
    Hop Limit: disabled
    Cost Limit: disabled
    Path-invalidation timeout: 10000 msec (default), Action: Tear
    AutoRoute: enabled  LockDown: disabled Loadshare: 200 [10000000] bw-based
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: dynamic path option 10 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled

  PCEP Info:
```

```
    Delegation state: Working: yes   Protect: no
    Current Path Info:
      Request status: processed
      Created via PCRep message from PCE server: 1.1.1.1
    Reported paths:
      Tunnel Name: csr551_t2001
       LSPs:
        LSP[0]:
         source 2.2.2.2, destination 7.7.7.7, tunnel ID 1, LSP ID 5
         State: Admin up, Operation active
         Setup type: SR
         Bandwidth: signaled 0
         LSP object:
           PLSP-ID 0x807D1, flags: D:0 S:0 R:0 A:1 O:2
         Reported path:
           Metric type: TE, Accumulated Metric 0

  History:
    Tunnel:
      Time since created: 34 minutes, 3 seconds
      Time since path change: 1 minutes, 44 seconds
      Number of LSP IDs (Tun_Instances) used: 5
    Current LSP: [ID: 5]
      Uptime: 1 minutes, 44 seconds
    Prior LSP: [ID: 3]
      ID: path option unknown
      Removal Trigger: path verification failed
  Tun_Instance: 5
  Segment-Routing Path Info (isis  level-1)
    Segment0[Node]: 3.3.3.3, Label: 20270
    Segment1[Node]: 6.6.6.6, Label: 20120
    Segment2[Node]: 7.7.7.7, Label: 20210
```

The following sample output is from the **show pce client lsp detail** command.

```
Device# show pce client lsp detail

PCC's tunnel database:
----------------------
Tunnel Name: d1_t1
 LSPs:
  LSP[0]:
   source 2.2.2.2, destination 7.7.7.7, tunnel ID 1, LSP ID 5
   State: Admin up, Operation active
   Setup type: SR
   Bandwidth: signaled 0
   LSP object:
     PLSP-ID 0x807D1, flags: D:0 S:0 R:0 A:1 O:2
   Reported path:
     Metric type: TE, Accumulated Metric 0
```

The following sample output is from the **show pce lsp detail** command which shows the tunnel is delegated.

```
Device# show pce lsp detail

Thu Jul  7 10:24:30.836 EDT

PCE's tunnel database:
----------------------
PCC 102.103.2.1:
```

```
Tunnel Name: d1_t1
 LSPs:
  LSP[0]:
   source 2.2.2.2, destination 7.7.7.7, tunnel ID 1, LSP ID 5
   State: Admin up, Operation active
   Binding SID: 0
   PCEP information:
     plsp-id 526289, flags: D:1 S:0 R:0 A:1 O:2
   Reported path:
     Metric type: TE, Accumulated Metric 0
      SID[0]: Node, Label 20270, Address 3.3.3.3
      SID[1]: Node, Label 20120, Address 6.6.6.6
      SID[2]: Node, Label 20210, Address 7.7.7.7
   Computed path:
     Metric type: TE, Accumulated Metric 30
      SID[0]: Node, Label 20270, Address 3.3.3.3
      SID[1]: Node, Label 20120, Address 6.6.6.6
      SID[2]: Node, Label 20210, Address 7.7.7.7
   Recorded path:
     None
```

The following sample output is from the **show pce client lsp detail** command for reported tunnel.

```
Device# show pce client lsp detail

PCC's tunnel database:
----------------------
Tunnel Name: d1_t2
 LSPs:
  LSP[0]:
   source 2.2.2.2, destination 7.7.7.7, tunnel ID 2, LSP ID 1
   State: Admin up, Operation active
   Setup type: SR
   Bandwidth: signaled 0
   LSP object:
     PLSP-ID 0x807D2, flags: D:0 S:0 R:0 A:1 O:2
   Reported path:
     Metric type: TE, Accumulated Metric 30
```

The following sample output is from the **show pce lsp detail** command which shows the tunnel is not delegated.

```
Device# show pce lsp detail

Thu Jul  7 10:29:48.754 EDT

PCE's tunnel database:
----------------------
PCC 10.0.0.1:

Tunnel Name: d1_t2
 LSPs:
  LSP[0]:
   source 2.2.2.2, destination 7.7.7.7, tunnel ID 2, LSP ID 1
   State: Admin up, Operation active
   Binding SID: 0
   PCEP information:
     plsp-id 526290, flags: D:0 S:0 R:0 A:1 O:2
   Reported path:
     Metric type: TE, Accumulated Metric 30
      SID[0]: Adj, Label 74, Address: local 172.16.0.1 remote 172.16.0.2
      SID[1]: Adj, Label 63, Address: local 173.17.0.1 remote 173.17.0.2
```

```
      SID[2]: Adj, Label 67, Address: local 174.18.0.1 remote 174.18.0.2
      SID[3]: Node, Label unknownAddress 7.7.7.7
   Computed path:
     None
   Recorded path:
     None
```

# Verifying Verbatim Path Option With Dynamic PCC

To verify proper operation with verbatim path option, use the following command:

```
R1#sh mpl tr tun tun 2
Name: R1_t2                                (Tunnel2) Destination: 66.66.66.66
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected
    path option 1, (SEGMENT-ROUTING) (PCE) type dynamic (verbatim) (Basis for Setup)

  Config Parameters:
    Bandwidth: 0        kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF
    Metric Type: TE (interface)
    Path Selection:
     Protection: any (default)
    Path-selection Tiebreaker:
      Global: not set   Tunnel Specific: not set   Effective: min-fill (default)
    Hop Limit: disabled [ignore: Verbatim Path Option]
    Cost Limit: disabled
    Path-invalidation timeout: 10000 msec (default), Action: Tear
    AutoRoute: disabled LockDown: disabled Loadshare: 0 [0] bw-based
    AutoRoute destination: enabled
    auto-bw: disabled
    Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
  Active Path Option Parameters:
    State: dynamic path option 1 is active
    BandwidthOverride: disabled  LockDown: disabled  Verbatim: enabled

  PCEP Info:
    Delegation state: Working: yes   Protect: no
    Delegation peer: 77.77.77.77
    Working Path Info:
      Request status: processed
      Created via PCRep message from PCE server: 77.77.77.77
      PCE metric: 4, type: TE
    Reported paths:
      Tunnel Name: Tunnel2_w
       LSPs:
        LSP[0]:
         source 11.11.11.11, destination 66.66.66.66, tunnel ID 2, LSP ID 1
         State: Admin up, Operation active
         Binding SID: 17
         Setup type: SR
         Bandwidth: requested 0, used 0
         LSP object:
           PLSP-ID 0x80002, flags: D:0 S:0 R:0 A:1 O:2
         ERO:
           SID[0]: Adj, Label 24, NAI: local 12.12.12.1 remote 12.12.12.2
           SID[1]: Adj, Label 26, NAI: local 25.25.25.2 remote 25.25.25.5
           SID[2]: Adj, Label 22, NAI: local 56.56.56.5 remote 56.56.56.6

  History:
    Tunnel:
```

```
      Time since created: 39 days, 19 hours, 9 minutes
      Time since path change: 1 minutes, 3 seconds
      Number of LSP IDs (Tun_Instances) used: 1
    Current LSP: [ID: 1]
      Uptime: 1 minutes, 3 seconds
  Tun_Instance: 1
  Segment-Routing Path Info (IGP information is not used)
    Segment0[Link]: 12.12.12.1 - 12.12.12.2, Label: 24
    Segment1[Link]: 25.25.25.2 - 25.25.25.5, Label: 26
    Segment2[Link]: 56.56.56.5 - 56.56.56.6, Label: 22
!
end
```

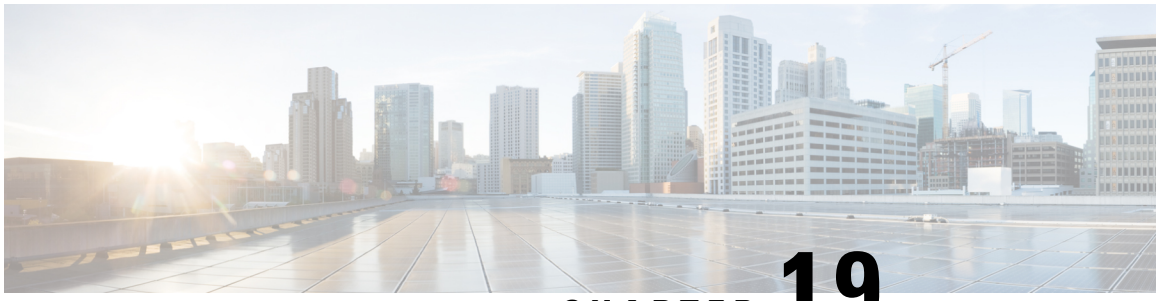
# Feature Information for Dynamic PCC

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

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

***Table 18: Feature Information for Dynamic PCC***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Dynamic PCC | Cisco IOS XE Amsterdam 17.3.2 | The Dynamic Path Computation Client (PCC) feature supports an LSP delegated to a Path Computation Element (PCE).Dynamic PCC aupports both RSVP-TE and SR-TE.<br>The following commands were added or modified:<br>**show pce client peer detail**, **show mpls traffic-eng tunnels tunnel 1**, **show pce client lsp detail**, **show pce lsp detail**. |



CHAPTER 19

# SR: PCE Initiated LSPs

The SR: PCE Initiated LSPs feature provides support for PCE-initiated LSPs in stateful PCE model on segment routing networks.


- Prerequisites for SR: PCE Initiated LSPs, on page 193
- Restrictions for SR: PCE Initiated LSPs, on page 193
- Information About SR: PCE Initiated LSPs, on page 193
- How to Configure SR: PCE Initiated LSPs, on page 195
- Additional References for SR: PCE Initiated LSPs, on page 201
- Feature Information for SR: PCE Initiated LSPs, on page 201


# Prerequisites for SR: PCE Initiated LSPs

- The Dynamic PCC feature must be configured.
- Auto tunnels must be enabled on the PCC.

# Restrictions for SR: PCE Initiated LSPs

- The SR: PCE Initiated LSPs feature supports only basic LSP generation and does not support TE attributes.

# Information About SR: PCE Initiated LSPs

## Overview of Path Computation Element Protocol

draft-ietf-pce-stateful-pce-21 describes Stateful Path Computation Element Protocol (PCEP) enables a router to report and optionally delegate Label Switched Paths (LSPs) which is established using either Resource Reservation Protocol (RSVP) protocol or Segment Routing Traffic Engineering (SR-TE) to a stateful Path Computation Element (PCE). An LSP delegated to a PCE can be updated by the PCE and a stateful PCE can compute and provide the path of an LSP to the Path Computation Client (PCC).

The **PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model (draft-ietf-pce-pce-initiated-lsp-11)** specifies a set of extensions to PCEP to enable stateful control of TE

LSPs across PCEP sessions in compliance with RFC4657. The **PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model** provides information about the following:

- Configuring LSPs on a PCC
- Delegating control of LSP to a PCE

# SR: PCE Initiated LSPs

The SR: PCE Initiated LSPs feature allows a client to create, setup, control, and delete an LSP from PCE server, which controls creating and deleting LSP on PCC through an PCE initiate message. PCE initiated LSP is automatically delegated to the PCE server that initiated the LSP. A PCE client processes an LSP initiate message. By using the LSP initiate message, PCE client can create or delete LSP.

When a failover occurs on a route processor (RP), the failover results in the RP being disconnected from the network. To reestablish the connection, the PCE server has to resend LSP initiate message to reclaim PCE Initiated LSPs on a client, else PCE initiated LSPs created by the client are automatically deleted.

You must use the **pce** command for establishing a PCEP session with PCC. The **force auto-route** command is used to advertise an LSP within an area via the autoroute announce message and across areas via the autoroute destination message. The decision to use autoroute announce or autoroute destination is performed by a device depending on the destination IP address. Enabling the **force auto-route** command for an initiated LSP allows automatic routing of traffic through a TE tunnel instead of routing traffic via manually configuring static routes. The autoroute announce message installs routes announced by the destination router and downstream routers into the routing table of a headend device that can be reached through a tunnel.

The PCC configuration includes IP addresses for each PCE (both primary and standby or more). The precedence for each PCE can be explicitly specified. If the precedence for two PCEs is same, PCE with smaller IP address has a higher precedence.

# Single and Redundant PCE Operations

The SR: PCE Initiated LSPs feature supports single and redundant PCE operations. In a single PCE operation, when a PCE fails, PCC waits until the state timeout expiry (60 seconds) to remove the LSP.

In a redundant PCE operation, if a Representational state transfer (REST) call is initiated to a standby PCE before the expiry of the timer, the initiated LSP is retained else, the LSP is removed.

**Note** The REST call must be initiated again to a standby PCE if the primary PCE fails, and the call must include the standby PCE IP address.

In a redundant PCE operation, PCC configurations include both primary and standby IP addresses for an LSP and the IP address with a lower precedence becomes the primary PCE. The IP addreseses are compared incase of equal priority.

# How to Configure SR: PCE Initiated LSPs

## Establishing a PCEP session with PCC

Perform this task to configure a PCEP session PCE server XR based XTC server.

```
configure terminal
pce
 address ipv4 192.0.2.1
end
```

The IP address 192.0.2.1 is the IP address of the transport controller.

## Advertising an LSP in a Network

```
configure terminal
 mpls traffic-eng pcc peer 192.0.2.1 source 203.0.113.1 force-autoroute
 end
```

In the above code snippet, 192.0.2.1 is PCE IP address and 203.0.113.1 is PCC source address for establishing a PCEP session.

## Specifying Precedence of a PCE for PCC

```
configure terminal
mpls traffic-eng pcc peer 192.0.2.1 source 203.0.113.1 force autoroute precedence 255
mpls traffic-eng pcc peer 192.0.2.2 source 203.0.113.1 force-autoroute precedence 100
end
```

In the above code snippet, 100 is a lower precedence than 255, which is the default precedence. Therefore, the device with IP address 192.0.2.2 becomes the primary PCE and the device with 192.0.2.1 becomes the standby PCE.

### Triggering PCE server precedence re-evaluation

A change in a PCE server's precedence is not considered a PCE server failure. So, the change in precedence does not trigger a redelegation timeout or a re-evaluation of LSP delegation to the PCE server at a PCC.

Re-evaluation of LSP delegation to PCE servers after CLI reconfiguration is controlled by the TE reoptimisation timer. By default, the TE reoptimisation timer is set to 3600 seconds.

You can accelerate the re-evaluation of LSP delegation from a PCC to PCE servers after you have changed the precedence of PCE servers or added new PCE servers. To do so, manually trigger TE reoptimisation using the following command in privileged EXEC mode:

```
mpls traffic-eng reoptimize
```

# Verifying LSP Configurations

## SUMMARY STEPS

1. **show pce ipv4 peer detail**
2. **show pce lsp detail**
3. **show pce client peer**
4. **show mpls traffic-eng tunnel tunnel** *number*

## DETAILED STEPS

### Step 1 show pce ipv4 peer detail

Use this command to verify PCEP session details on a PCE. In this example, the term instantiation indicates that PCE supports initiated LSP.

```
Device# show pce ipv4 peer detail


PCE's peer database:

--------------------

Peer address: 52.2.2.2----' PCC IP address

               State: Up

  Capabilities: Stateful, Segment-Routing, Update, Instantiation
```

### Step 2 show pce lsp detail

Use this command to verify the initiated LSP on a PCE.

```
Device# show pce lsp detail


PCE's tunnel database:

----------------------

PCC 52.2.2.2 ----' PCC IP address


Tunnel Name: Test1--------' tunnel name set by REST Call

LSPs:
```

```
LSP[0]:
 source 52.2.2.2, destination 57.7.7.7, tunnel ID 2000, LSP ID 1
 State: Admin up, Operation active
 Binding SID: 26
 PCEP information:
   plsp-id 526288, flags: D:1 S:0 R:0 A:1 O:2 C:1
 LSP Role: Single LSP
 State-sync PCE: None
 PCC: 52.2.2.2
 LSP is subdelegated to: None
 Reported path:
   Metric type: TE, Accumulated Metric 2
    SID[0]: Adj, Label 25, Address: local 102.105.3.1 remote 102.105.3.2
    SID[1]: Adj, Label 24, Address: local 104.105.8.2 remote 104.105.8.1
    SID[2]: Adj, Label 38, Address: local 104.107.10.1 remote 104.107.10.2
 Computed path: (Local PCE)
   None
   Computed Time: Not computed yet
 Recorded path:
   None
 Disjoint Group Information:
   None
```

**Step 3** **show pce client peer**

Use this command to verify a PCEP session output on a PCC and to verify if the **force-autoroute** command is enabled.

```
Device# show pce client peer


PCC's peer database:

------------------------------

Peer address: 51.1.1.1, Precedence: 255

  State up

  Capabilities: Stateful, Update, Segment-Routing,Force-autoroute
```

**Step 4** **show mpls traffic-eng tunnel tunnel** *number*

Use this command to verify the output of the initiated LSP tunnel on a PCC.

```
Device# show mpls traffic-eng tunnel tunnel 2000


Name: Test1                                (Tunnel2000) Destination: 57.7.7.7 Ifhandle: 0x11E
(auto-tunnel for pce client)

  Status:

    Admin: up         Oper: up     Path: valid       Signalling: connected

    path option 1, (SEGMENT-ROUTING) (PCE) type dynamic (Basis for Setup)


  Config Parameters:

    Bandwidth: 0        kbps (Global)  Priority: 7  7   Affinity: 0x0/0xFFFF

    Metric Type: TE (default)

    Path Selection:

     Protection: any (default)

    Path-selection Tiebreaker:

      Global: not set   Tunnel Specific: not set   Effective: min-fill (default)

    Hop Limit: disabled

    Cost Limit: disabled
```

```
  Path-invalidation timeout: 10000 msec (default), Action: Tear
  AutoRoute: enabled  LockDown: disabled Loadshare: 0 [0] bw-based
  auto-bw: disabled
  Fault-OAM: disabled, Wrap-Protection: disabled, Wrap-Capable: No
Active Path Option Parameters:
  State: dynamic path option 1 is active
  BandwidthOverride: disabled  LockDown: disabled  Verbatim: disabled

PCEP Info:
  Delegation state: Working: yes   Protect: no
  Delegation peer: 51.1.1.1
  Working Path Info:
    Request status: delegated
    SRP-ID: 1
    Created via PCInitiate message from PCE server: 51.1.1.1-----' IP address
    PCE metric: 2, type: TE
  Reported paths:
    Tunnel Name: Test1
     LSPs:
      LSP[0]:
       source 52.2.2.2, destination 57.7.7.7, tunnel ID 2000, LSP ID 1
       State: Admin up, Operation active
      Binding SID: 26
       Setup type: SR
       Bandwidth: requested 0, used 0
```

```
        LSP object:

          PLSP-ID 0x807D0, flags: D:0 S:0 R:0 A:1 O:2

        Metric type: TE, Accumulated Metric 2

        ERO:

          SID[0]: Adj, Label 25, NAI: local 102.105.3.1 remote 102.105.3.2

          SID[1]: Adj, Label 24, NAI: local 104.105.8.2 remote 104.105.8.1

          SID[2]: Adj, Label 38, NAI: local 104.107.10.1 remote 104.107.10.2

      PLSP Event History (most recent first):

        Mon Jul 17 08:55:04.448: PCRpt update LSP-ID:1, SRP-ID:1, PST:1, METRIC_TYPE:2, REQ_BW:0,
USED_BW:0

        Mon Jul 17 08:55:04.436: PCRpt create LSP-ID:1, SRP-ID:1, PST:1, METRIC_TYPE:2, REQ_BW:0,
USED_BW:0

  History:

    Tunnel:

      Time since created: 2 hours, 42 minutes

      Time since path change: 2 hours, 42 minutes

      Number of LSP IDs (Tun_Instances) used: 1

    Current LSP: [ID: 1]

      Uptime: 2 hours, 42 minutes

  Tun_Instance: 1

  Segment-Routing Path Info (isis  level-2)

    Segment0[Link]: 102.105.3.1 - 102.105.3.2, Label: 25

    Segment1[Link]: 104.105.8.2 - 104.105.8.1, Label: 24

    Segment2[Link]: 104.107.10.1 - 104.107.10.2, Label: 38
```

# Additional References for SR: PCE Initiated LSPs

### Standards and RFCs

| Standard/RFC | Title |
|---|---|
| draft-ietf-pce-pce-initiated-lsp-11 | *PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model* |
| RFC 5440 | *Path Computation Element (PCE) Communication Protocol (PCEP)* |
| RFC 8231 | *Path Computation Element (PCE) Communication Protocol Generic Requirements* |

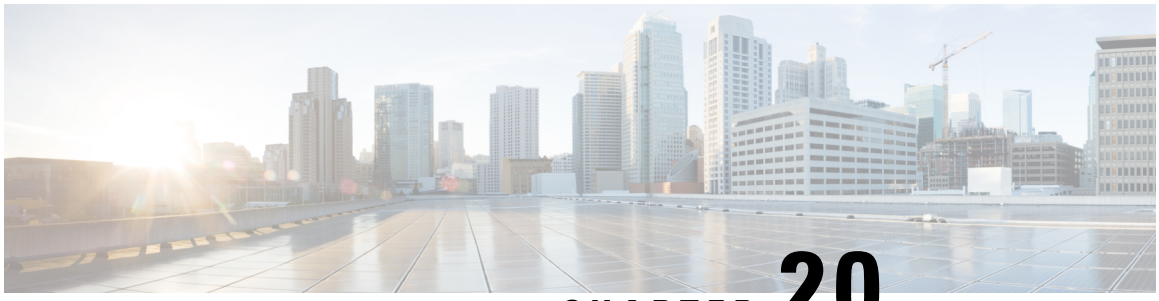
# Feature Information for SR: PCE Initiated LSPs

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

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

***Table 19: Feature Information for SR: PCE Initiated LSPs***

| Feature Name | Releases | Feature Information |
|---|---|---|
| SR: PCE Initiated LSPs | Cisco IOS XE Amsterdam 17.3.2 | The SR: PCE Initiated LSPs provides support for PCE-initiated LSPs in stateful PCE model on segment routing networks.<br>The following commands were introduced or modified: **mpls traffic-eng pcc**, **pce**, **show mpls traffic-eng tunnel**, **show pce client peer**, **show pce ipv4 peer**, **show pce lsp**. |



CHAPTER 20

# ISIS - SR: uLoop Avoidance

The ISIS - SR: uLoop Avoidance feature extends the ISIS Local Microloop Protection feature thereby preventing the occurrences of microloops during network convergence after a link-down event or link-up event.


- Prerequisites for ISIS - SR: uLoop Avoidance, on page 203
- Restrictions for ISIS - SR: uLoop Avoidance, on page 203
- Information About ISIS - SR: uLoop Avoidance, on page 203
- How to Enable ISIS - SR: uLoop Avoidance, on page 207
- Additional References for ISIS - SR: uLoop Avoidance, on page 208
- Feature Information for ISIS - SR: uLoop Avoidance, on page 208


# Prerequisites for ISIS - SR: uLoop Avoidance

- The ISIS - SR: uLoop Avoidance feature is disabled by default. When the Topology-Independent Loop-Free Alternate (TI-LFA) feature is configured, this feature is enabled automatically. See the "Topology-Independent LFA" section in the *Using Segment Routing with IS-IS* module for more information.

# Restrictions for ISIS - SR: uLoop Avoidance

- The ISIS - SR: uLoop Avoidance feature supports 2-node on the same subnet on a LAN network.

# Information About ISIS - SR: uLoop Avoidance

## Microloops

When changes occur in a network topology because of the failure or restoration of a link or a network device, IP Fast Reroute enables rapid network convergence by moving traffic to precomputed backup paths until regular convergence mechanisms move traffic to a newly computed best path, also known as a post-convergence path. This network convergence may cause short microloops between two directly or indirectly connected devices in the topology. Microloops are caused when different nodes in the network calculate alternate paths

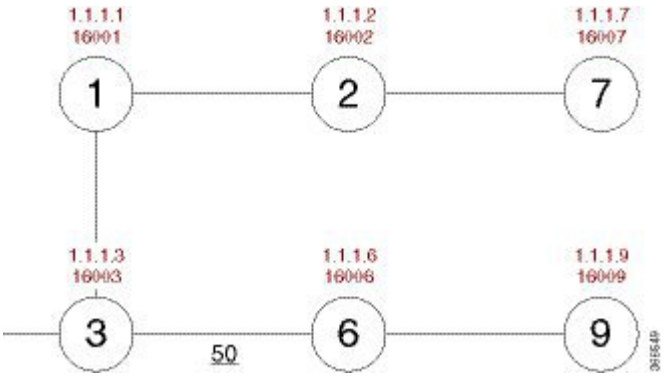
at different times and independently of each other. For instance, if a node converges and sends traffic to a neighbor node, which has not converged yet, traffic may loop between the two nodes.

Microloops may or may not result in traffic loss. If the duration of a microloop is short, that is the network converges quickly, packets may loop for a short duration before their time-to-live (TTL) expires. Eventually, the packets will get forwarded to the destination. If the duration of the microloop is long, that is one of the routers in the network is slow to converge, packets may expire their TTL or the packet rate may exceed the bandwidth, or the packets might be out of order, and packets may get dropped.

Microloops that are formed between a failed device and its neighbors are called local uloops, whereas microloops that are formed between devices that are multiple hops away are called remote uloops. Local uloops are usually seen in networks where local loop-free alternate (LFA) path is not available. In such networks, remote LFAs provide backup paths for the network.

The information discussed above can be illustrated with the help of an example topology as shown in the following figure.

***Figure 21: Microloop Example Topology***



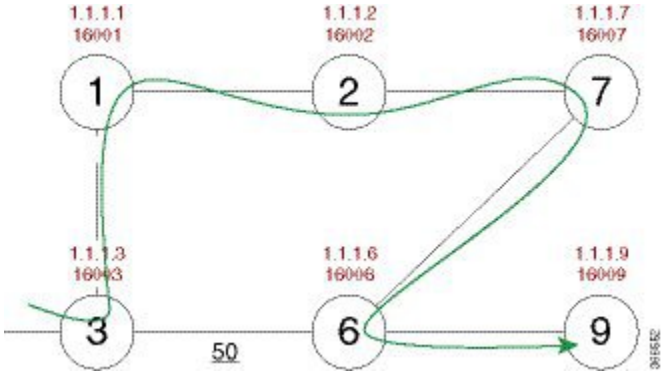

The assumptions in this example are as follows:

- The default metrics is 10 for each link except for the link between Node 3 and Node 6, which has a metric of 50. The order of convergence with SPF backoff delays on each node is as follows:
  - Node 3—50 milliseconds
  - Node 1—500 milliseconds
  - Node 2—1 second
  - Node 2—1.5 seconds

A packet sent from Node 3 to Node 9, the destination, traverses via Node 6.

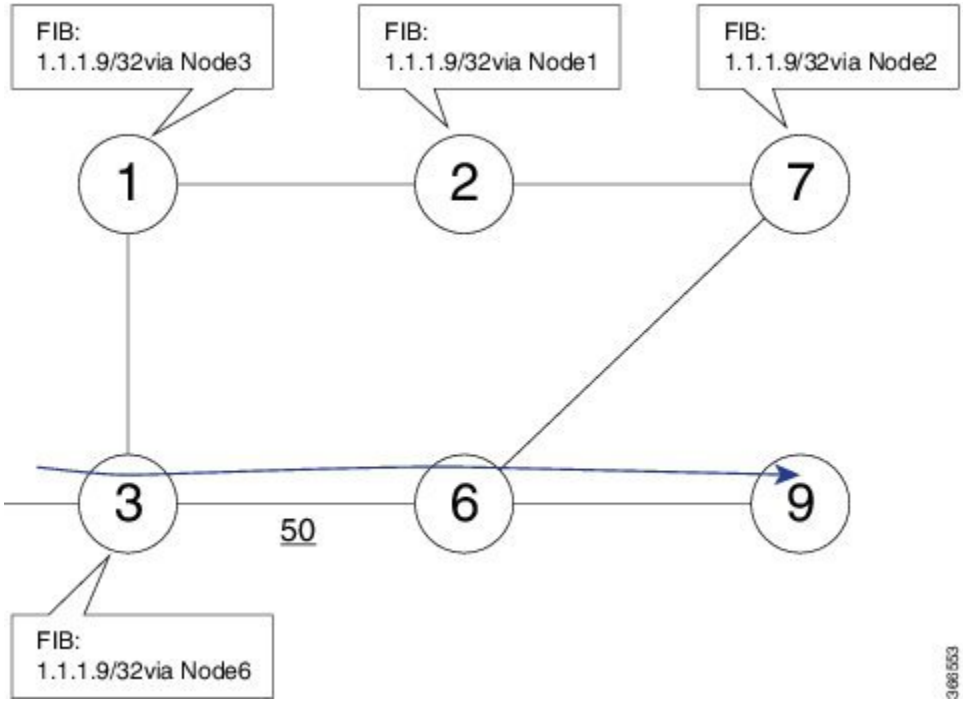
If a link is established between Node 6 and Node 7, the shortest path for a packet from Node 3 to Node 9 would be Node 1, Node 2, Node 7, and Node 6 before the packet reaches the destination, Node 9.

*Figure 22: Microloop Example Topology—Shortest Path*



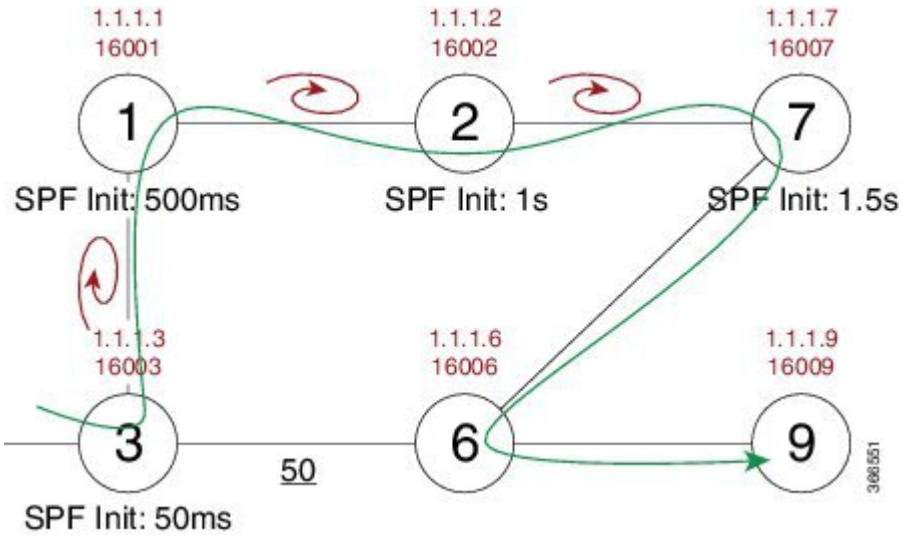

The following figure shows the Forwarding Information Base (FIB) table in each node before the link between Node 6 and Node 7 is established. The FIB entry contains the prefix of the destination node (Node 9) and the next hop.

*Figure 23: Microloop Example Topology—FIB Entry*




When the link between Node 6 and Node 7 comes up, microloops occur for the links based on the order of convergence of each node. In this example, Node 3 converges first with Node 1 resulting in a microloop between Node 3 and Node 1. Then, Node 1 converges next resulting in a microloop between Node 1 and Node 2. Next, Node 2 converges next resulting in a microloop between Node 2 and Node 7. Finally, Node 7 converges resolving the microloop and the packet reaches the destination Node 9, as shown in the following figure.

*Figure 24: Microloop Example Topology—Microloops*




Adding the SPF convergence delay, microloop results in a loss of connectivity for 1.5 seconds, which is the convergence duration specified for node 7.

# Segment Routing and Microloops

The ISIS - SR: uLoop Avoidance feature supports the following scenarios:

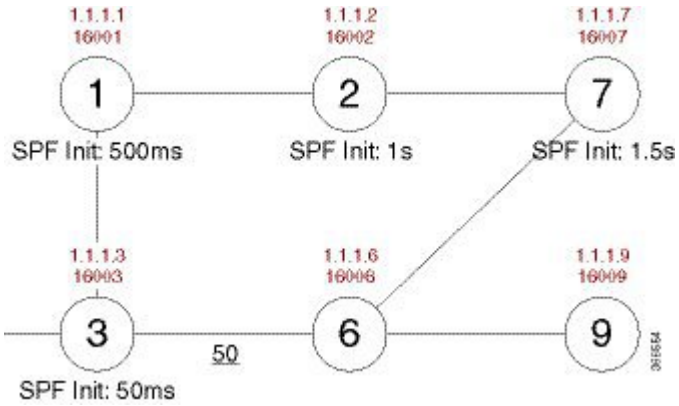
- Link-up or link-down for point-to-point links and a LAN segment with two nodes
- Link cost decrease or increase when a node is up or down due to the overload bit being set or unset

The **microloop avoidance segment-routing** command must be enabled on a node to prevent microloops.

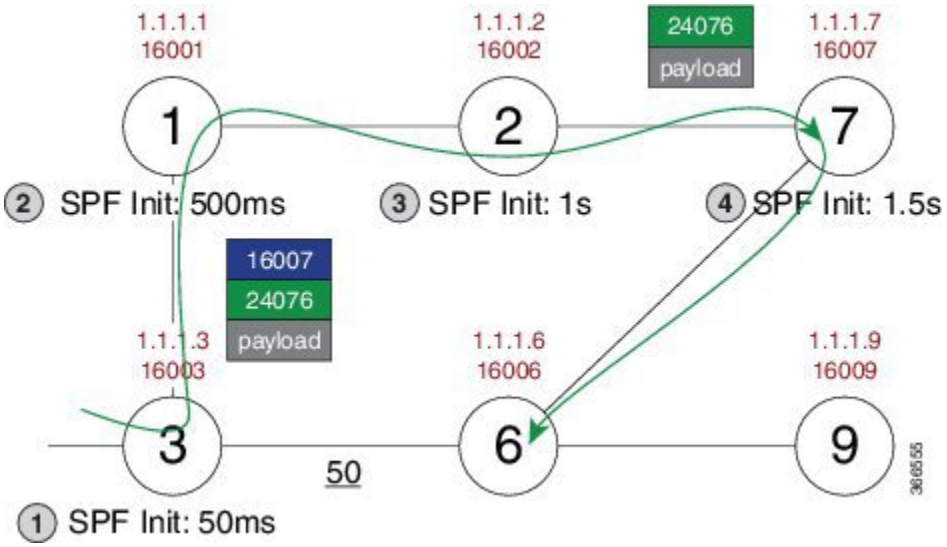
## How Segment Routing Prevents Microloops?

Using the example used to explain microloops, this section explains how to segment routing prevents microloops. Node 3 in the example is enabled with the **microloop avoidance segment-routing** command. After the link between Node 6 and Node 7 comes up, Node 3 computes a new microloop on the network.

*Figure 25: Microloop Example Topology—Segment Routing*



Instead of updating the FIB table, Node 3 builds a dynamic loop-free alternate (LFA) SR TE policy for the destination (Node 9) using a list of segments IDs, which include the prefix segment ID (SID) of Node 7, which is 16007, and the adjacency segment ID (SID) of Node 6, which is 24076.




So, the SR TE policy enables a packet from Node 3 reaches its destination Node 9, without the risk of microloop until the network converges. Finally, Node 3 updates the FIB for the new path.

Use the protected keyword with the **microloop avoidance segment-routing** command, to enable microloop avoidance for protected prefixes only. The **microloop avoidance rib-update-delay** *milliseconds* command can be used to configure the delay in milliseconds for a node to wait before updating the node's forwarding table and stop using the microloop avoidance policy. The default value for the RIB delay is 5000 milliseconds.

# How to Enable ISIS - SR: uLoop Avoidance

## Enabling Microloop Avoidance

The following is a sample configuration code snippet to enable microloop avoidance.

```
router isis
 fast-reroute per-prefix level-2 all
 microloop avoidance segment-routing
 microloop avoidance rib-update-delay 3000
```

## Verifying Microloop Avoidance

Use the **show isis rib** and **show ip route** commands to check if the repair path exists or not.

```
Router# show isis rib 20.20.20.0 255.255.255.0

IPv4 local RIB for IS-IS process sr

IPV4 unicast topology base (TID 0, TOPOID 0x0) =================
Repair path attributes:
 DS - Downstream, LC - Linecard-Disjoint, NP - Node-Protecting
 PP - Primary-Path, SR - SRLG-Disjoint
```

```
20.20.20.0/24 prefix attr X:0 R:0 N:0 prefix SID index 2 - Bound (ULOOP EP)
 [115/L2/130] via 77.77.77.77(MPLS-SR-Tunnel5), from 44.44.44.44, tag 0,
LSP[2/5/29]
 prefix attr: X:0 R:0 N:0
 SRGB: 16000, range: 8000 prefix-SID index: None
 (ULOOP_EP)(installed)
 - - - - - -
 [115/L2/130] via 16.16.16.6(Ethernet2/0), from 44.44.44.44, tag 0, LSP[2/5/29]
 prefix attr: X:0 R:0 N:0
 SRGB: 16000, range: 8000 prefix-SID index: None
 (ALT)
```

```
Router# show ip route 20.20.20.0

Routing entry for 20.20.20.0/24
 Known via "isis", distance 115, metric 130, type level-2
 Redistributing via isis sr
 Last update from 77.77.77.77 on MPLS-SR-Tunnel5, 00:00:43 ago
 SR Incoming Label: 16002 via SRMS
 Routing Descriptor Blocks:
 * 77.77.77.77, from 44.44.44.44, 00:00:43 ago, via MPLS-SR-Tunnel5,
 * prefer-non-rib-labels, merge-labels
 Route metric is 130, traffic share count is 1
 MPLS label: 16002
 MPLS Flags: NSF
```

# Additional References for ISIS - SR: uLoop Avoidance

### Related Documents

| Related Topic | Document Title |
|---|---|
| Segment Routing and IS-IS | *Using Segment Routing with IS-IS* |
| Overview of IS-IS concepts | "IS-IS Overview and Basic Configuration" module in the *IP Routing: ISIS Configuration Guide* |
| ISIS Local Microloop Protection | "ISIS Local Microloop Protection" module in the *IP Routing: ISIS Configuration Guide* |

### Standards/RFCs

| Standard/RFC | Title |
|---|---|
| draft-francois-rtgwg-segment-routing-uloop-00 | *Loop avoidance using Segment Routing* |

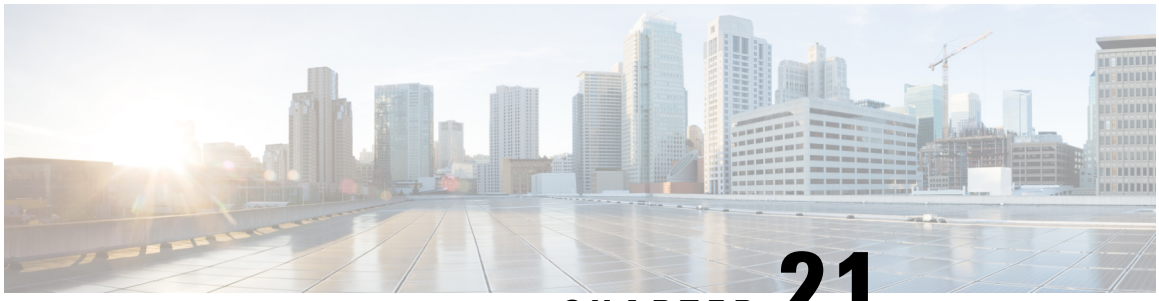
# Feature Information for ISIS - SR: uLoop Avoidance

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

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

*Table 20: Feature Information for ISIS - SR: uLoop Avoidance*

| Feature Name | Releases | Feature Information |
|---|---|---|
| ISIS - SR: uLoop Avoidance | Cisco IOS XE Amsterdam 17.3.2 | The ISIS - SR: uLoop Avoidance feature extends the ISIS Local Microloop Protection feature thereby preventing the occurrences of microloops during network convergence after a link-down event or link-up event.<br><br>The following commands were introduced or modified: **microloop avoidance**, **microloop avoidance rib-update-delay**, **show mpls traffic tunnel**. |



CHAPTER 21

# BGP - SR: BGP Prefix SID Redistribution

The BGP - SR: BGP Prefix SID Redistribution feature provides support for BGP Prefix-SID in IPv4 prefixes in segment routing—BGP networks.


- Prerequisites for BGP - SR: BGP Prefix SID Redistribution, on page 211
- Information About BGP - SR: BGP Prefix SID Redistribution, on page 211
- How to Enable BGP - SR: BGP Prefix SID Redistribution, on page 212
- Additional References for BGP - SR: BGP Prefix SID Redistribution, on page 214
- Feature Information for BGP - SR: BGP Prefix SID Redistribution, on page 214


# Prerequisites for BGP - SR: BGP Prefix SID Redistribution

- Mulitprotocol Label Switching (MPLS) must be configured.

# Information About BGP - SR: BGP Prefix SID Redistribution

## Segment Routing and BGP

Segment Routing uses Multiprotocol Label Switching (MPLS) labels to create a path to guide a packet in a network. Using segment routing, an MPLS label range is reserved with MPLS Forwarding Infrastructure (MFI). This label range is called Segment Routing Global Block (SRGB). A prefix SID assigned to a prefix is an extension of SRGB.

To support segment routing, Border Gateway Protocol (BGP) requires the ability to advertise a segment identifier (SID) for a BGP prefix. A BGP-Prefix-SID is the segment identifier of the BGP prefix segment in an segment routing with BGP network. A BGP-Prefix-SID is also an instruction to forward the packet over an ECMP-aware best-path computed by BGP to a related prefix. When BGP nodes communicate with neighbor nodes in a network, the BGP Update, message sent to neighbor nodes, includes the Prefix-SID Label in Labeled Unicast NLRI and a prefix SID index in a new attribute called Prefix SID attribute.

To support forwarding paths for traffic engineering, the forwarding path may need to be different from the optimal path. Hence, each BGP node assigns a local label to the neighbors and advertises the local label as adjacency SID through BGP--link state updates.

The BGP - SR: BGP Prefix SID Redistribution feature can be enabled by using the **connected-prefix-sid-map** command in the segment routing MPLS configuration mode. Additionally, you also need to enable the **segment-routing mpls** command in the router configuration mode for each address family.

**Note** In Cisco IOS XE Everest 16.6.1, IPv4 prefixes only are supported.

## Segment Routing for Locally Sourced Routes

Interface host routes configured on local nodes are known as locally sourced routes. If segment routing is enabled, a BGP node includes the explicit or implicit null as prefix SID label and prefix SID attribute and advertises the prefix to a neighbor node.

If explicit-null is not configured on a neighbor, the MPLS Implicit Null label (3) is advertised to a neighbor node. If explicit-null is configured on a neighbor, the MPLS Explicit Null label corresponding to the address family of the prefix is advertised (0 for IPv4) to a neighbor node.

## Segment Routing for Received Prefixes

BGP nodes that receive prefix SID attribute from a neighbor node via communication, add the label in the outgoing label as the prefix when a route is added to the RIB. The local label and prefix SID index is included in the RIB.

## Segment Routing for Redistributed Routes

A source protocol on a BGP node allocates local label depending on the received prefix SID index and SRGB available on a local node. A source protocol provides the prefix SID index and the derived local label to RIB. BGP uses the local label from RIB as a label in the Labeled Unicast update sent to neighbors nodes.

## BGP--MFI Interaction

BGP registers with MFI as a client and binds the label derived from SID index and SRGB as local label (with which traffic is expected to arrive) for the prefix.

# How to Enable BGP - SR: BGP Prefix SID Redistribution

## Enabling BGP-Prefix-SID

```
segment-routing mpls
 connected-prefix-sid-map */--------> Configures Prefix to SIDIndex Map that can be queried
 by BGP/IGP /*
 address-family ipv4
 10.0.0.1/255.0.0.0 index 10 range 11.0.0.1
```

# Enabling BGP for Segment Routing

```
router bgp 2
 address-family-ipv4
  segment-routing mpls
```

# Verifying BGP - SR: BGP Prefix SID Redistribution

This section shows how to verify the BGP - SR: BGP Prefix SID Redistribution feature with the help of an example network, in which, a device configured with segment routing is connected to two devices configured with Border Gateway Protocol (BGP). In each device, the **show segment-routing mpls** command is used to view the configuration.

The following is configuration on the device configured with segment routing.

```
segment-routing mpls
global-block 10000 13000
!
connected-prefix-sid-map
  address-family ipv4
   12.1.1.1/32 index 3 range 1
  exit-address-family
!
  segment-routing mpls


interface Loopback0
ip address 12.1.1.1 255.255.255.255

router bgp 1
neighbor 10.1.1.2 remote-as 2
!
address-family ipv4
  redistribute connected
  segment-routing mpls
  neighbor 10.1.1.2 activate
  neighbor 10.1.1.2 send-label
exit-address-family
```

The following is the configuration on the first device configured with BGP.

```
segment-routing mpls

router bgp 2
neighbor 10.1.1.1 remote-as 1
neighbor 11.1.1.2 remote-as 3
!
address-family ipv4
  redistribute connected
  neighbor 10.1.1.1 activate
  neighbor 10.1.1.1 send-label
  neighbor 11.1.1.2 activate
  neighbor 11.1.1.2 send-label
exit-address-family
```

The following is the configuration on the second device configured with BGP.

```
segment-routing mpls

router bgp 3
neighbor 11.1.1.1 remote-as 2
!
address-family ipv4
```

```
  redistribute connected
  neighbor 11.1.1.1 activate
  neighbor 11.1.1.1 send-label
exit-address-family
```

# Additional References for BGP - SR: BGP Prefix SID Redistribution

### Related Documents

### Standards and RFCs

| Standard/RFC | Title |
|---|---|
| RFC3107 | *Carrying Label Information in BGP-4* |

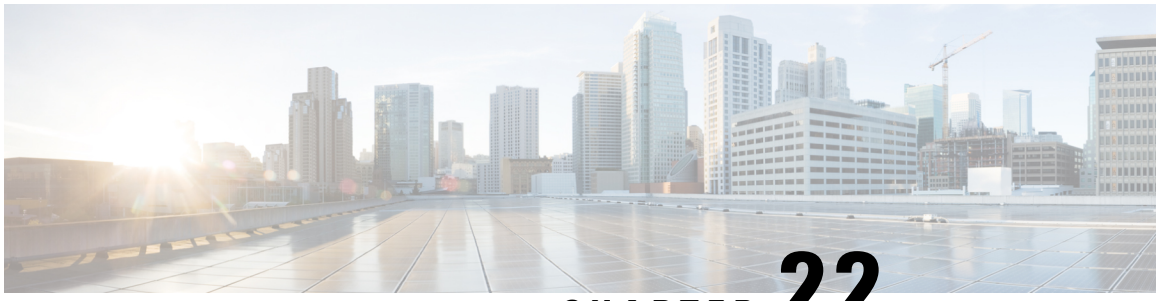
# Feature Information for BGP - SR: BGP Prefix SID Redistribution

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

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

***Table 21: Feature Information for BGP - SR: BGP Prefix SID Redistribution***

| Feature Name | Releases | Feature Information |
|---|---|---|
| BGP - SR: BGP Prefix SID Redistribution | Cisco IOS XE Amsterdam 17.3.2 | The BGP - SR: BGP Prefix SID Redistribution feature provides support for BGP Prefix-SID in IPv4 prefixes in segment routing—BGP networks.<br>The following commands were introduced or modified: **connected-prefix-sid-map**, **segment-routing**. |



CHAPTER 22

# Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS

In a Segment Routing (SR) enabled network a centralized controller that programs SR tunnels needs to know the Maximum Segment Identifier (SID) Depth (MSD) supported by the head-end at node and/or link granularity to push the SID stack of an appropriate depth. MSD is relevant to the head-end of a SR tunnel or binding-SID anchor node where binding-SID expansions might result in creation of a new SID stack.


- Restrictions for Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS, on page 215
- Information About Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS, on page 215
- Verifying Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS , on page 217
- Feature Information for Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS, on page 218


# Restrictions for Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS

- In IOS-XE as there no line cards, link-MSD is not advertised.

# Information About Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS

**Note** This feature is enabled by default and no specific configuration is required to enable this functionality.

## Maximum SID Depth

You can use IGPs to signal the MSD of a node or link to a centralized controller by:

- Advertising node-MSD to its peers.
- Providing the MSD information to BGP-LS.

Path Computation Element Protocol (PCEP) SR extensions signal MSD in SR PCE capability TLV and metric object. However, if PCEP is not supported/configured on the head-end of a SR tunnel or a binding-SID anchor node and controller does not participate in IGP routing, it has no way to learn the MSD of nodes. BGP-LS defines a way to expose topology and associated attributes and capabilities of the nodes in that topology to a centralized controller. Typically, BGP-LS is configured on a small number of nodes that do not necessarily act as head-ends. In order for BGP-LS to signal MSD for all the SR capable nodes in the network, MSD capabilities should be advertised by every IGP router in the network.

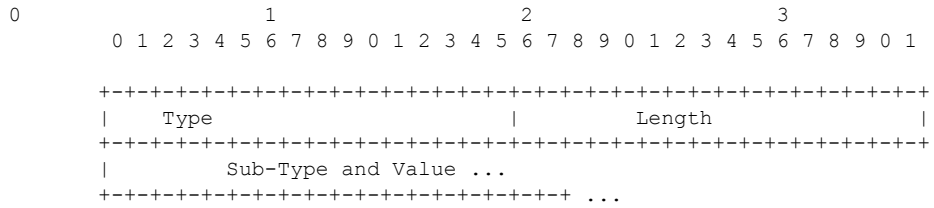
Readable Label Depth Capability (RLDC) is used by a head-end to insert Entropy Label (EL) at appropriate depth, so it could be read by transit nodes. MSD in contrary signals ability to push SID's stack of a particular depth.

MSD of type 1 (IANA registry) is used to signal the number of SIDs a node is capable of imposing to be used by a path computation element/controller. It is only relevant to the part of the stack created as the result of the computation. MSD advertises the total number of labels that a node is capable of imposing regardless of the number of service labels.

# Node Maximum SID Depth Advertisement

A new Type/Length/Value (TLV) within the body called node MSD TLV is defined to carry the provisioned SID depth of the router originating the Router Information (RI) Link State Advertisement (LSA). Node MSD is the lowest MSD supported by the node.

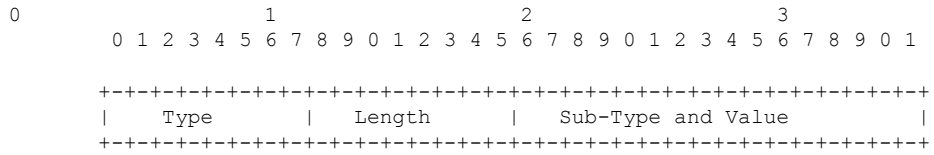
### Node Maximum SID Depth Advertisement for OSPF




The Type (2 bytes) of this TLV is 12 (that is the suggested value to be assigned by IANA). Length is variable (minimum of 2, multiple of 2 octets) and represents the total length of value field. Value field consists of a 1 octet sub-type (IANA Registry) and 1 octet value.

Sub-Type 1, MSD and the value field contains maximum MSD of the device originating the RI LSA. Node maximum MSD falls in the range of 0-254. 0 represents lack of the ability to push MSD of any depth; any other value represents that of the node. This value should represent the lowest value supported by node.

### Node Maximum SID Depth Advertisement for IS-IS




Node MSD is a sub-TLV for TLV 242. The type of this sub-TLV is 23. Length is variable (minimum of 2, multiple of 2 octets).

Sub-Type 1, MSD and the value field contains maximum MSD of the device originating the RI LSA. Node maximum MSD falls in the range of 0-254. 0 represents lack of the ability to push MSD of any depth; any other value represents that of the node. This value should represent the lowest value supported by node.

## Getting the Node MSD from Hardware

IS-IS and OSPF are updated about the maximum SID Depth for the node from the underlying hardware. Based on that IS-IS and OSPF update the value in its TLVs.

## Advertising the MSD to BGP-LS

lGP sends the information to LSLIB to make the MSD information available to BGP-LS. It can be node MSD or link MSD information. You also need to configure **distribute linkstate** under IS-IS for MSD to work. Perform the following steps to configure distribute link-state:

```
Device# configure terminal
Device(config)# router isis
Device(config-router)# distribute link-state
```

# Verifying Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS

### Verifying Advertise Maximum SID Depth Using IS-IS

The following show command is used to verify the node MSD TLV:

```
Device# show isis database verbose
Router CAP:   10.10.10.1, D:0, S:0
    Segment Routing: I:1 V:0, SRGB Base: 16000 Range: 8000
       Segment Routing Algorithms: SPF, Strict-SPF
      Router CAP:   2.2.2.2, D:0, S:0
    Segment Routing: I:1 V:0, SRGB Base: 16000 Range: 8000
    Segment Routing Algorithms: SPF, Strict-SPF
    Node-MSD
      MSD: 16
```

### Verifying Advertise Maximum SID Depth Using OSPF

The following show command is used to verify the node MSD TLV:

```
Device# show ip ospf database opaque-area type router-information
TLV Type: Segment Routing Node MSD
   Length: 2
     Sub-type: Node Max Sid Depth, Value: 16
```

# Feature Information for Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS

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

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

***Table 22: Feature Information for Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Advertise Maximum SID Depth by IS-IS and OSPF to BGP-LS | Cisco IOS XE Amsterdam 17.3.2 | In a Segment Routing (SR) enabled network a centralized controller that programs SR tunnels needs to know the Maximum Segment Identifier (SID) Depth (MSD) supported by the head-end at node and/or link granularity to push the SID stack of an appropriate depth. MSD is relevant to the head-end of a SR tunnel or binding-SID anchor node where binding-SID expansions might result in creation of a new SID stack.<br><br>The following commands were introduced or modified by this feature: **distribute link-state**, **show isis database verbose**, **show ip ospf database opaque-area type router-information**. |

CHAPTER 23

# RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel

This document describes support for link protection also referred as next-hop (NHOP) protection using the backup Segment-Routing Traffic Engineering (SR-TE) autotunnel. It protect the links over which the RSVP Traffic Engineering (RSVP-TE) tunnel traverses.


- Feature Information for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel, on page 219
- Prerequisites for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel, on page 220
- Restrictions for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel, on page 220
- Information About RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel, on page 221
- How to Configure RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel, on page 223
- Verifying RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel, on page 225


## Feature Information for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel

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

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

*Table 23: Feature Information for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel*

| Feature Name | Releases | Feature Information |
|---|---|---|
| RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel | Cisco IOS XE Amsterdam 17.3.2 | This feature provides support for link protection also referred as next-hop (NHOP) protection using the backup Segment-Routing Traffic Engineering (SR-TE) autotunnel. It protect the links over which the RSVP Traffic Engineering (RSVP-TE) tunnel traverses.<br>The following commands were introduced by this feature: **ip explicit-path name path1 enable**, **show mpls traffic-eng tunnels tunnel 65436**, **show ip explicit-paths**, **show mpls traffic-eng tunnels tunnel 65436 \| show Segment-Routing Path Info**, **show mpls traffic-eng fast-reroute database**, **show ip rsvp fast-reroute  sh mpls traffic-eng auto-tunnel backup**. |

# Prerequisites for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel

Before enabling SR-TE backup autotunnel, ensure that the following technologies are configured in your setup:

- IS-IS Network Point to Point Interfaces
- Segment Routing

Additionally, prior knowledge of the following technologies are required:

- MPLS Traffic-Engineering
- RSVP Traffic-Engineering
- Fast reroute

# Restrictions for RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel

- SR-TE backup autotunnel cannot be used for bandwidth protection.
- SR-TE backup autotunnel can only be used as a backup for RSVP-TE tunnel protection.

# Information About RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel

## Benefits of RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel

With increased complexity in the network, scalability becomes a challenge due to maintenance of RSVP-TE tunnels with complex signaling as well as high overhead on routers within the network. Backup autotunnel feature can help reduce the complexity in a Segment Routing (SR) network. Autotunnel backup feature has the following benefits:

- Backup tunnels are built automatically hence eliminating the need for users to pre-configure each backup tunnel and then assign the backup tunnel to the protected interface.
- With the backup tunnels configured, area of protection gets expanded. Fast reroute (FRR) neither protects IP traffic nor LDP labels that do not use TE tunnel.
- Backup SR-TE autotunnel allows additional means of migration to SR network without disrupting the existing traffic passing through RSVP-TE tunnels.

## Backup AutoTunnel

Backup autotunnels on a router helps to build dynamic backup tunnels whenever required. This prevents creating of static SR-TE tunnels.

To protect a label-switched path (LSP) in the absence of static SR-TE tunnels, you need to do the following:

- Preconfigure each backup tunnel.
- Assign the backup tunnels to the protected interfaces.

An LSP requests backup protection from Resource Reservation Protocol (RSVP) FRR in the following situations:

- Receipt of the first RSVP Resv message.
- Receipt of an RSVP path message with the protection attribute after the LSP has been established without protection attribute.
- Detection of changed Record Route Object (RRO).

If there is no backup tunnel protecting the interface used by the LSP, the LSP remained unprotected. Some of the reasons why a backup tunnel may not be available are:

- Static backup tunnels are not configured.
- Static backup tunnels are configured, but may not be able to protect the LSP because there is not enough bandwidth available, or the tunnel protects a different pool, or the tunnel is not available.

If a backup tunnel is not available, the following two backup tunnels are created dynamically:

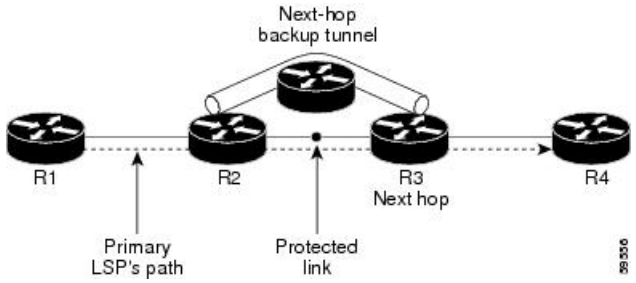
- NHOP—Protects against link failure.
- NNHOP—Protects against node failure.

**Note** At the penultimate hop, only an NHOP backup tunnel is created.

### Link Protection

Backup tunnels that bypass only a single link of the LSP's path provide link protection. They protect LSPs if a link along their path fails by rerouting the LSP's traffic to the next hop (bypassing the failed link). These are referred to as NHOP backup tunnels because they terminate at the LSP's next hop beyond the point of failure.

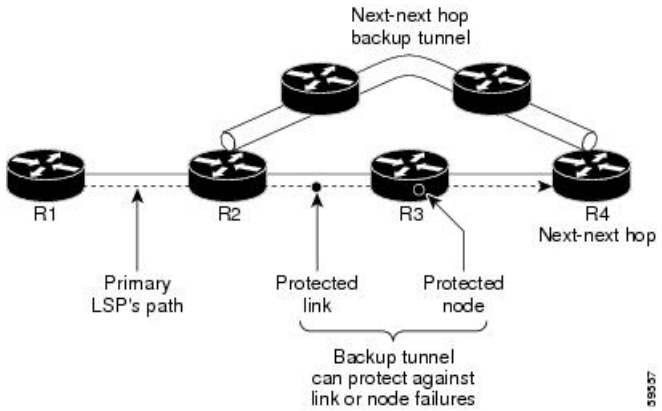
***Figure 26: Next-Hop Backup Tunnel***




### Node Protection

Backup tunnels that bypass next-hop nodes along LSP paths are called NNHOP backup tunnels because they terminate at the node following the next-hop node of the LSPs, thereby bypassing the next-hop node. They protect LSPs by enabling the node upstream of a link or node failure to reroute the LSPs and their traffic around the failure to the next-hop node. NNHOP backup tunnels also provide protection from link failures because they bypass the failed link and the node.

***Figure 27: Next-Next Hop Backup Tunnel***



### Explicit Paths

Explicit paths are used to create backup autotunnels as follows:

- NHOP excludes the protected link's IP address.
- NNHOP excludes the NHOP router ID.
- The explicit-path name is _auto-tunnel_tunnel *xxx*, where *xxx* matches the dynamically created backup tunnel ID.

### Range for Backup AutoTunnels

You can configure the tunnel range for backup autotunnels. By default, the last 100 TE tunnel IDs are used, which is 65,436 to 65,535. Autotunnels detect tunnel IDs that are alloted starting with the lowest number.

For example, if you configure a tunnel within the range of 1000 to 1100. And statically configured TE tunnel also falls in the same range then routers do not use those IDs. If those static tunnels are removed, the MPLS-TE dynamic tunnel software can use those IDs.

# How to Configure RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel

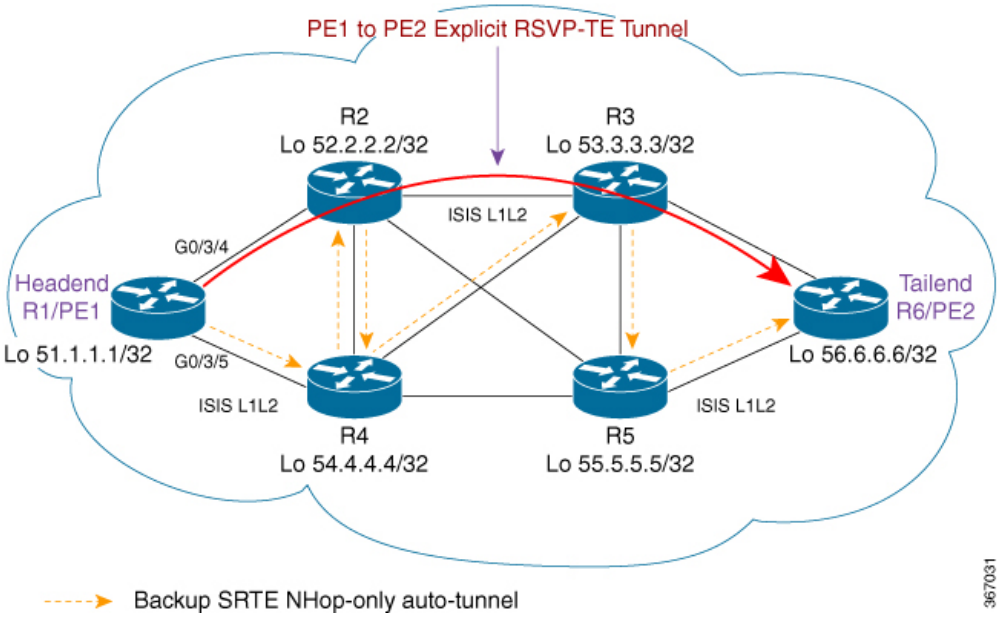
## Configuring Explicit Path for Point-to-Point Network Type

For SR-TE autotunnel backup feature to work interfaces have to be point-to-point network type.

```
interface Loopback0
 ip address 51.1.1.1 255.255.255.255
 ip router isis 1
end
!
interface GigabitEthernet0/2/0
 ip address 101.102.6.1 255.255.255.0
 ip router isis 1
 negotiation auto
 mpls traffic-eng tunnels
 isis network point-to-point
 ip rsvp bandwidth
end
!
interface GigabitEthernet0/2/4
 ip address 101.104.1.1 255.255.255.0
 ip router isis 1
 negotiation auto
 mpls traffic-eng tunnels
 isis network point-to-point
 ip rsvp bandwidth
end
```

# Configuring Explicit RSVP-TE Tunnel With FRR

*Figure 28: Explicit RSVP-TE Tunnel*




1. Configure explicit path from R1/PE1 to R6/PE2 that traverses through the routers R2 and R3.

```
ip explicit-path name path1 enable
 index 1 next-address 209.165.202.128
 index 2 next-address 209.165.201.0
 index 3 next-address 192.168.0.0
 index 4 next-address 209.165.200.224
```

2. Configure explicit RSVP-TE tunnel.

```
interface Tunnel1
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 209.165.200.224
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng path-option 10 explicit name path1
 tunnel mpls traffic-eng record-route
end
```

3. Configure the primary RSVP-TE Tunnel 1 with FRR to activate the protection process.

```
interface tunnel 1
 tunnel mpls traffic-eng fast-reroute
```

4. Configure the global command to enable link protection using SR-TE autotunnel.

```
mpls traffic-eng auto-tunnel backup segment-routing nhop-only
```

**Note** This command needs to be available in all the nodes that require link protection.

The Primary RSVP-TE tunnel need to be protected that gets initialized from headend R1/PE1 to destination R6/PE2 and traversing through next node R2 and so on. In this case, R1/PE1 is the Point of Local Repair (PLR) and R2 is the Mid-Point (MP). With link protection, the SR-TE Backup AutoTunnel provides protection to the link from R1/PE1 to R2 by traversing through the path R1/PE1 -> R4 and R4 -> R2, hence converging back to the MP.

# Verifying RSVP-TE Protection using Segment Routing Traffic Engineering AutoTunnel

Use the **show interfaces Tunnel** command to verify if SR-TE AutoTunnel is generated and up.

```
Device#show interfaces Tunnel65436
Tunnel65436 is up, line protocol is up
```

Use the **show mpls traffic-eng tunnels** command to verify if the backup AutoTunnel is a SR-TE Tunnel.

```
Device#show mpls traffic-eng tunnels tunnel 65436
Name: R1_t65436 (Tunnel65436) Destination: 209.165.201.0
 Status:
  Admin: up Oper: up Path: valid Signalling: connected
  path option 1, (SEGMENT-ROUTING) type explicit __dynamic_tunnel65436 (Basis for
Setup, path weight 20)
```

Use the **show ip explicit-paths** command to verify if the SR-TE Backup Tunnel is using a secondary path to reach the node.

```
Device#show ip explicit-paths
PATH __dynamic_tunnel65436 (strict source route, path complete, generation 49, status
non-configured)
1: exclude-address 101.102.5.1
```

Use the **show mpls traffic-eng tunnels tunnel 65436 | s Segment-Routing Path Info** command to verify if the backup tunnel is going through the path R1/PE1 to R4 and finally to destination R2 which is the mid-point.

```
Device#show mpls traffic-eng tunnels tunnel 65436 | s Segment-Routing Path Info
Segment-Routing Path Info (isis level-1)
 Segment0[Link]: 101.104.1.1 - 101.104.1.2, Label: 19
 Segment1[Link]: 102.104.6.2 - 102.104.6.1, Label: 18
```

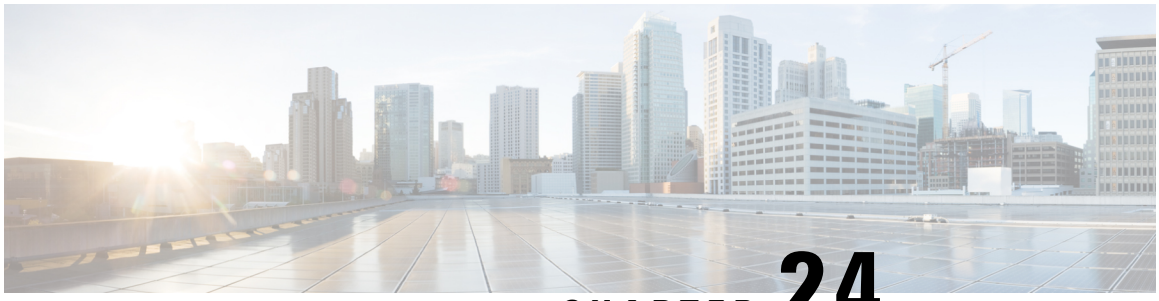
Use the **show mpls traffic-eng auto-tunnel backup** command to verify if the auto-tunnel backup state is correct.

```
Device#show mpls traffic-eng auto-tunnel backup
State: Enabled
 Auto backup tunnels: 1 (up: 1, down: 0)
 Tunnel ID Range: 65436 - 65535
```

```
 Create Nhop Only: Yes
 Check for deletion of unused tunnels every: 3600 Sec
 SRLG: Not configured

Config:
 unnumbered-interface: Loopback0
  Affinity/Mask: 0x0/0xFFFF
```

Use the **show mpls traffic-eng fast-reroute database** command to verify if the primary link through which the RSVP-TE LSP is traversing is protected.

```
Device#show mpls traffic-eng fast-reroute database
P2P Headend FRR information:
Protected tunnel In-label Out intf/label FRR intf/label Status
-------------------------- -------- -------------- -------------- ------
Tunnel1 Tun hd Gi0/3/4:30 Tu65436:30 ready


Device#show ip rsvp fast-reroute
P2P Protect BW Backup
Protected LSP I/F BPS:Type Tunnel:Label State Level Type
------------- ------- -------- ------------- ------ ----- ------
R1_t1 Gi0/3/4 0:G Tu65436:28 Ready any-unl Nhop
```



CHAPTER 24

# ISIS Manual Adjacency SID

The Integrated Intermediate System-to-Intermediate System (IS-IS) manual adjacency SID feature provides information about manually provisioned Adjency SIDs.


- Feature Information for ISIS Manual Adjacency SID, on page 227
- Information About ISIS Manual Adjacency SID, on page 227
- Configuring Manual Adjacency SID, on page 229
- Verifying Manual Adjacency SID, on page 230


## Feature Information for ISIS Manual Adjacency SID

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

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

***Table 24: Feature Information for ISIS Manual Adjacency SID***

| Feature Name | Releases | Feature Information |
|---|---|---|
| ISIS Manual Adjacency SID | Cisco IOS XE Amsterdam 17.3.2 | The Integrated Intermediate System-to-Intermediate System (IS-IS) manual adjacency SID feature provides information about manually provisioned Adjency SIDs.<br>The following commands were introduced by this feature:<br>**adjacency-sid [absolute \| index]**<*value*> **[protected]**.<br>. |

## Information About ISIS Manual Adjacency SID

Segment routing (SR) networks often use SR Traffic Engineering (SR-TE) to influence the path the specific traffic takes over the network. SR-TE tunnels can be provisioned manually on the tunnel head, but often they are calculated and provisioned by the central controller. In many cases operator of the network wants to be able to force the traffic over specific nodes and links.

To force the traffic over a certain node in the SR network operators can use Prefix-SID that is advertised by the node. Many times the anycast Prefix SID is used which forces the traffic to go over specific location where multiple nodes share the same Prefix-SID.

To force the traffic over the specific link, an Adjacency-SID (Adj-SID) is used. The problem with the existing implementation of the Adj-SID is that it is a dynamically allocated value which is in contrast to manually provisioned prefix-SID. The fact that the Adj-SID is dynamically allocated brings a set of problems:

- The value is not persistent over reload or process restart.
- The value is not known upfront so controller cannot use it unless it has access to the information flooded by IGP (natively or through BGP-LS).
- Each link is allocated a unique adj-SID value which prevents the same adj-SID to be shared by multiple links.

To address the above mentioned issues, the adj-SIDs are enhanced and now thay are capable of the following:

- Support manually provisioned adj-SID that is persistent over reload and restart.
- Support same adj-SID to be provisioned for multiple adjacencies to the same neighbor.
- Support same adj-SID to be provisioned for multiple adjacencies going to different neighbors.
- Multiple manual Adj-SIDs can be configured for a single adjacency.

# Manual Adjacency SID

The existing IS-IS Adj-SID infrastructure that is being used for dynamically allocated Adj-SIDs is extended to support the new persistent Adj-SID requirements. A new CLI command is also introduced to manually assign Adj-SID values for point-to-point links. Multiple Adj-SIDs can be provisioned on a single point-to-point interface. Same Adj-SID can be provisioned on multiple point-to-point interfaces leading to the same or different neighbors.

All manual Adj-SIDs are assigned from a range of labels called Segment Routing Local Block(SRLB). The default SRLB Range is 15000-15999.

Manual Adj-SIDs can be configured as an Index or an Absolute value. If it is configured as an index, the absolute label is calculated as an index + SRLB starting label. For example, if you configure 56 as a manual Adj-SID index, the absolute label would be 15000 + 56 = 15056. If it is configured as an absolute, the label itself is the absolute value. For example, if you configure 56 as an absolute manual Adj-SID, the absolute label would be 56 only. Labels (both index and absolute) can be configured as protected or non-protected. By default, all the labels are non-protected.

# Adjacency SID Advertisement

Manual adj-SIDs are advertised using existing ISIS adj-SID sub-TLV as defined in the ISIS SR extension draft. If the same value of the adj-SID has been provisioned on multiple interfaces, the S-Flag is set in the adj-SID sub-TLV. In the case of manual adj-SID, P flag is always set.

If the provisioned adj-SID has been configured as protected, the B-flag also gets set.

Adjacency-SIDs are always advertised as a label value and never as an index even if the index are used to configure the adj-SID.

# Adjacency SID Forwarding

When the adj-SID value is only configured on a single interface, then the ISIS installs forwarding entries for manually allocated adj-SIDs. The primary path for any Adj-SID is a POP operation over the point-to-point interface for which the Adj-SID is allocated. If the allocated adj-SID is eligible for backup and the backup path is available, IS-IS programs the backup path as well. The backup path for Adj-SID is equal to the backup path computed for the neighbor router-id address.

If the same adj-SID value is configured on multiple links forwarding happens as the following:

- Primary path with POP operation is installed via each link where adj-SID is configured with that value.
- For each primary path if the adj-SID is configured as protected on the primary interface and backup is available, backup path gets installed. Backup path is represented as a backup path associated with the neighbor router-id address.

# Configuration Prerequisites

- Ensure that segment routing is configured globally.
- Ensure that segment routing is configured using IS-IS.

# Configuring Manual Adjacency SID

```
Device#configure terminal
Device(config)#interface ethernet0/1
Device(config-if)#isis adjacency-sid [absolute | index] <value> [protected]
```

**[index]** – (Optional) It is used if the adjacency SID is configured as an index to the SRLB range. If the index keyword is not used the value is expected to represent the absolute value of the label.

**[absolute]** - (Optional) It is used if the adjacency SID is configured as absolute value.

**<*value*>** - It represents the adj-SID label value or index. For the adj-SID to be programmed and advertised, the value/index must fall in the valid SRLB range.

**[protected]** - (Optional) It is used to protect the manual adj-SIDs. By default, manual adj-SIDs are not protected.

### Modifying Segment Routing Local Block (SRLB) Range

```
Device#configure terminal
Device(config)#segment-routing mpls
Device(config-srmpls)#local-block 7000 7999
```

# Verifying Manual Adjacency SID

### Verifying Label in SR APP Database

```
Device#show segment-routing mpls lb assigned-sids
Adjacency SID Database
 C=> In conflict
 S=> Shared
 R=> In range
SID  STATE       PROTOCOL     TOPOID      LAN      PRO NEIGHBOR    INTERFACE
15378 R
                 ISIS         0           N        N   10.0.0.3     Ethernet0/1
```

### Verifying Label in MPLS Forwarding

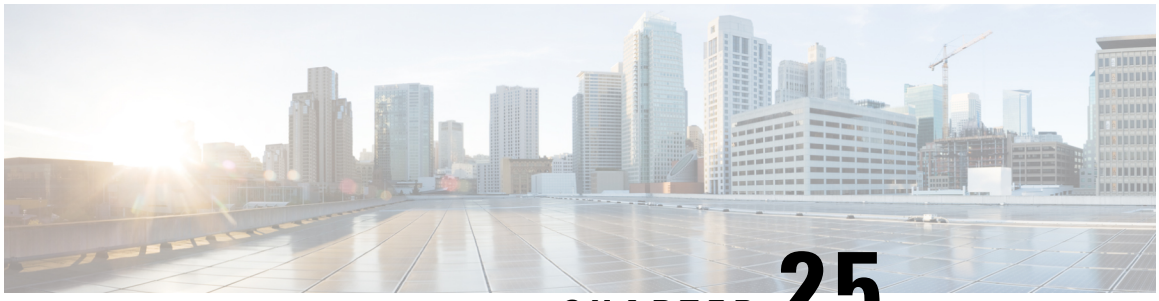
```
       Device# show mpls forwarding-table
       Local      Outgoing        Prefix               Bytes Label      Outgoing
Next Hop
       Label      Label           or Tunnel Id         Switched         interface
       15378      Pop Label       0.0.60.18-A          0                Et0/0
10.0.0.2 □== Configured only for interface e0/0
```

### Verifying Shared Label

```
       Device# show mpls forwarding-table
       Local      Outgoing       Prefix               Bytes Label      Outgoing        Next
 Hop
       Label      Label          or Tunnel Id         Switched         interface
       15378      Pop Label      0.0.60.18-A          0                Et0/0
10.0.0.2  □== Same Label is configured for 2 interfaces
                  Pop Label      0.0.60.18-A          0                Et0/1
10.0.0.3  □==
```

### Verifying ISIS LSP

```
       Device# sh isis database verbose R1.00-00
       xxxxxx
       xxxxxx
       Adjacency SID Value:15378 F:0 B:0 V:1 L:1 S:1 P:1 Weight:0 □== P (Persistent) flag
is always 1 if it is Manual Adj-SID
       xxxxxx

       P -> Persistent Flag (0 for Dynamic Adj-SID and 1 for Manual Adj-SID)
       S -> Shared Flag (1 if label is shared by multiple adjacencies)
```



CHAPTER 25

# OSPF Manual Adjacency SID

The OSPF manual adjacency SID feature supports configuration of static adjacency SIDs for Segment Routing with OSPFv2.


- Feature Information for OSPF Manual Adjacency SID, on page 231
- Information About OSPF Manual Adjacency SID, on page 231
- How to Configure OSPF Manual Adjacency SID, on page 233


## Feature Information for OSPF Manual Adjacency SID

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

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

***Table 25: Feature Information for OSPF Manual Adjacency SID***

| Feature Name | Releases | Feature Information |
|---|---|---|
| OSPF Manual Adjacency SID | Cisco IOS XE Amsterdam 17.3.2 | The OSPF manual adjacency SID feature supports configuration of static adjacency SIDs for Segment Routing with OSPFv2.<br>The following command was introduced by this feature:<br>**adjacency-sid index** *value* [**protected**] |

## Information About OSPF Manual Adjacency SID

Segment routing (SR) networks often use SR Traffic Engineering (SR-TE) to influence the path specific traffic takes over the network. SR-TE tunnels can be provisioned manually on the tunnel headend, or are calculated and provisioned by a central controller.

For traffic engineering, operators of a network need to be able to force traffic over specific nodes and links. To force traffic over a certain node in the SR network, operators can use the Prefix SID that is advertised by

the node. An anycast Prefix SID can be used to route traffic to specific node when multiple nodes advertise the same Prefix SID.

To force traffic through a certain link, operators can use the adjacency SID of the link. Without the support for manually-configured adjacency SIDs, adjacency SIDs are dynamically allocated. Dynamically allocated SIDs have the following disadvantages in relation to traffic engineering:

- The dynamic value is not persistent over reload or process restart.
- The dynamic value is not known upfront, and so a controller cannot use it unless it has access to the information flooded by IGP (natively or through BGP-LS).
- Each link is allocated a unique Adjacent SID value. With such an allocation, the same adjacency SID cannot be allocated to multiple links.

The OSPF Manual Adjacency SID feature introduces support for manually-configured adjacency SIDs. With manually-configured static adjacency SIDs,

- provisioned adjacency SID is persistent over reload and restart.
- multiple adjacency SIDs can be configured for a single adjacency.

# Prerequisites for OSPF Manual Adjacency SID

- Segment Routing must be configured globally.
- Segment Routing must be configured for the OSPF instance.

# Restrictions for OSPF Manual Adjacency SID

- Static adjacency SIDs can be configured only for point-to-point links and not for broadcast links.
- Do not assign the same adjacency SID to multiple links. Group adjacency SIDs are not supported.
- Do not configure the same static adjacency SID in multiple IGPs or IGP instances. Such a configuration is not supported and a conflict handling mechanism for the scenario is yet to be implemented.
- Specify static adjacency SIDs as an indices to the Segment Routing Local Block (SRLB). Static adjacency SIDs cannot be specified as absolute values of labels in the SRLB.

# Manual Adjacency SIDs

Static adjacency SIDs can be configured for point-to-point links with OSPFv2.

Manual adjacency SIDs must be assigned from the SRLB. The default range of SRLB labels is 15000 to 15999. You can modify the SRLB range using the **local-block** *range-start range-end* command.

You can assign static adjacency SIDs as indices to the SRLB. Based on the index assigned, the label for the adjacency SID is calculated as `label = SRLB_range_start + index_value`.

By default, static adjacency SIDs are not protected, and therefore, you can specify whether a static adjacency SID must be protected or not during configuration.

## Manual Adjacency SID Advertisement

Static adjacency SIDs are advertised using the existing Adj-SID Sub-TLV of the Extended Link LSA as defined in OSPF Extensions for Segment Routing.

For static adjacency SIDs, the P-flag (Persistent flag) is set in the Adj-SID Sub-TLV.

If a static adjacency SID is protected, then the B-flag is set in the Adj-SID Sub-TLV.

Static adjacency SIDs are always advertised as labels. When the static adjacency SID is configured as an index, the absolute value of the label is calculated and the label value is advertised.

## Manual Adjacency SID Forwarding

When a static adjacency SID is configured for a point-to-point interface, OSPFv2 installs forwarding entries for the manually allocated adjacency SID. The primary path for an adjacency SID is a POP operation over the point-to-point interface for which the adjacency SID is allocated.

If the manually-allocated adjacency SID is eligible for backup and a backup path is available, OSPFv2 programs the backup path as well. The backup path for a manually-allocated adjacency SID is the backup path computed for the neighbor router.

# How to Configure OSPF Manual Adjacency SID

## Modifying Segment Routing Local Block Range

```
Device#configure terminal
Device(config)#segment-routing mpls
Device(config-srmpls)#local-block range-start range-end
```

*range-start* and *range-start* indicate the modified range bounds for Segment Routing Local Block (SRLB).

OSPF advertises the SRLB in the SR Local Block TLV of the Router Information (R.I.) Opaque LSA.

Only a single range is supported for SRLB. If an SR Local Block TLV has multiple ranges, the receiving router ignores the TLV.

```
Device#configure terminal
Device(config)#segment-routing mpls
Device(config-srmpls)#local-block 7000 7999
```

## Configuring OSPF Manual Adjacency SID

```
Device#configure terminal
Device(config)#interface <interface>
Device(config-if)#ip ospf adjacency-sid index <sid_value> [protected]
```

`<sid_value>` must be an index to the SRLB. Configuration of the adjacency SID as an absolute label value is yet to be supported.

`[protected]` (Optional) – This keyword is used to protect a manual adjacency SID. By default, manual adjacency SIDs are not protected.

# Verifying OSPF Manual Adjacency SID

You can verify SIDs assigned to adjacencies and whether an SID is static or dynamic using the commands **show ip ospf segment-routing adjacency-sid** and **show ip ospf segment-routing adjacency-sid detail**. The output of either command also shows additional information such as the neighbor linked through an adjacency, whether an adjacency is protected or not, and the backup next-hop and interface for a protected adjacency.

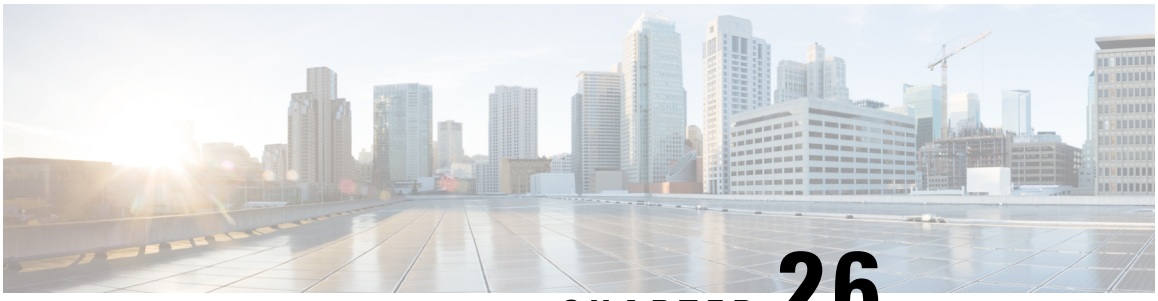
- `router#show ip ospf segment-routing adjacency-sid`

```
            OSPF Router with ID (2.0.0.0) (Process ID 1)
    Flags: S - Static, D - Dynamic,  P - Protected, U - Unprotected, G - Group, L -
Adjacency Lost

Adj-Sid  Neighbor ID     Interface          Neighbor Addr   Flags   Backup Nexthop
Backup Interface
-------- --------------- ------------------ --------------- ------- ---------------
------------------
16       3.0.0.0         Et0/2.3            3.3.3.3         D U
17       3.0.0.0         Et0/2.1            2.3.1.3         D U
24       3.0.0.0         Et0/2.1            2.3.1.3         D P     2.3.2.3
Et0/2.2
25       1.0.0.0         Et0/0              1.2.0.1         D U
26       1.0.0.0         Et0/0              1.2.0.1         D P     2.3.1.3
Et0/2.1
27       3.0.0.0         Et0/2.2            2.3.2.3         D U
28       3.0.0.0         Et0/2              2.3.0.3         D U
29       3.0.0.0         Et0/2              2.3.0.3         D P     2.4.0.4
Et0/1
30       4.0.0.0         Et0/1              2.4.0.4         D U
34       4.0.0.0         Et0/1              2.4.0.4         D P     2.3.1.3
Et0/2.1
15010    1.0.0.0         Et0/0              1.2.0.1         S P     2.3.1.3
Et0/2.1
15210    1.0.0.0         Et0/0              1.2.0.1         S U
15230    3.0.0.0         Et0/2              2.3.0.3         S P     2.4.0.4
Et0/1
15240    4.0.0.0         Et0/1              2.4.0.4         S U
15800    3.0.0.0         Et0/2.1            2.3.1.3         S U
15801    3.0.0.0         Et0/2.2            2.3.2.3         S U
15802    3.0.0.0         Et0/2.3            3.3.3.3         S U
15810    3.0.0.0         Et0/2.1            2.3.1.3         S P     2.3.2.3
Et0/2.2
```

- `router#show ip ospf segment-routing adjacency-sid detail`

```
            OSPF Router with ID (2.0.0.0) (Process ID 1)
Label 16, Paths 1, Dynamic
   Nbr id 3.0.0.0, via 3.3.3.3 on Et0/2.3, Unprotected
Label 17, Paths 1, Dynamic
   Nbr id 3.0.0.0, via 2.3.1.3 on Et0/2.1, Unprotected
Label 24, Paths 1, Dynamic
   Nbr id 3.0.0.0, via 2.3.1.3 on Et0/2.1, Protected, Nbr Prefix 33.33.33.33
      Primary path: via 2.3.1.3 on Et0/2.1, out-label 3
      Repair path: via 2.3.2.3 on Et0/2.2, out-label 3, cost 31, labels 0
Label 25, Paths 1, Dynamic
   Nbr id 1.0.0.0, via 1.2.0.1 on Et0/0, Unprotected
Label 26, Paths 1, Dynamic
   Nbr id 1.0.0.0, via 1.2.0.1 on Et0/0, Protected, Nbr Prefix 1.1.1.1
      Primary path: via 1.2.0.1 on Et0/0, out-label 3
      Repair path: via 2.3.1.3 on Et0/2.1, out-label 16001, cost 31, labels 0
Label 27, Paths 1, Dynamic
   Nbr id 3.0.0.0, via 2.3.2.3 on Et0/2.2, Unprotected
Label 28, Paths 1, Dynamic
```

```
   Nbr id 3.0.0.0, via 2.3.0.3 on Et0/2, Unprotected
Label 29, Paths 1, Dynamic
   Nbr id 3.0.0.0, via 2.3.0.3 on Et0/2, Protected, Nbr Prefix 3.3.3.3
      Primary path: via 2.3.0.3 on Et0/2, out-label 3
      Repair path: via 2.4.0.4 on Et0/1, out-label 16003, cost 21, labels 0
Label 30, Paths 1, Dynamic
   Nbr id 4.0.0.0, via 2.4.0.4 on Et0/1, Unprotected
Label 34, Paths 1, Dynamic
   Nbr id 4.0.0.0, via 2.4.0.4 on Et0/1, Protected, Nbr Prefix 4.4.4.4
      Primary path: via 2.4.0.4 on Et0/1, out-label 3
      Repair path: via 2.3.1.3 on Et0/2.1, out-label 16004, cost 31, labels 0
Label 15010, Paths 1, Static
   Nbr id 1.0.0.0, via 1.2.0.1 on Et0/0, Protected, Nbr Prefix 1.1.1.1
      Primary path: via 1.2.0.1 on Et0/0, out-label 3
      Repair path: via 2.3.1.3 on Et0/2.1, out-label 16001, cost 31, labels 0
Label 15210, Paths 1, Static
   Nbr id 1.0.0.0, via 1.2.0.1 on Et0/0, Unprotected
Label 15230, Paths 1, Static
   Nbr id 3.0.0.0, via 2.3.0.3 on Et0/2, Protected, Nbr Prefix 3.3.3.3
      Primary path: via 2.3.0.3 on Et0/2, out-label 3
      Repair path: via 2.4.0.4 on Et0/1, out-label 16003, cost 21, labels 0
Label 15240, Paths 1, Static
   Nbr id 4.0.0.0, via 2.4.0.4 on Et0/1, Unprotected
Label 15800, Paths 1, Static
   Nbr id 3.0.0.0, via 2.3.1.3 on Et0/2.1, Unprotected
Label 15801, Paths 1, Static
   Nbr id 3.0.0.0, via 2.3.2.3 on Et0/2.2, Unprotected
Label 15802, Paths 1, Static
   Nbr id 3.0.0.0, via 3.3.3.3 on Et0/2.3, Unprotected
Label 15810, Paths 1, Static
   Nbr id 3.0.0.0, via 2.3.1.3 on Et0/2.1, Protected, Nbr Prefix 33.33.33.33
      Primary path: via 2.3.1.3 on Et0/2.1, out-label 3
      Repair path: via 2.3.2.3 on Et0/2.2, out-label 3, cost 31, labels 0
```



CHAPTER 26

# OSPFv2 Segment Routing Strict SPF

The OSPFv2 Segment Routing Strict Shortest Path First (SPF) feature provides information about the strict SPF segment identifiers (SIDs).


- Feature Information for OSPFv2 Segment Routing Strict SPF, on page 237
- Restrictions for OSPFv2 Segment Routing Strict SPF, on page 238
- Information About OSPFv2 Segment Routing Strict SPF, on page 238
- Enabling and Disabling OSPFv2 Segment Routing Strict SPF, on page 239
- Configuring OSPFv2 Segment Routing Strict SPF SID, on page 240
- Verifying OSPFv2 Segment Routing Strict SPF, on page 240


# Feature Information for OSPFv2 Segment Routing Strict SPF

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

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

***Table 26: Feature Information for OSPFv2 Segment Routing Strict SPF***

| Feature Name | Releases | Feature Information |
|---|---|---|
| OSPFv2 Segment Routing Strict SPF | Cisco IOS XE Amsterdam 17.3.2 | The OSPFv2 Segment Routing Strict SPF feature provides the provision to support strict shortest path algorithm. It mandates that the packets are forwarded according to SPF algorithm and instructs any router in the path to ignore any possible local policy overriding the SPF decision.<br>The following commands were added or modified:<br>**address-family ipv4 strict-spf**. |

# Restrictions for OSPFv2 Segment Routing Strict SPF

- All the nodes in an OSPF area must be strict SPF capable and each node must have at least one strict SPF SID for the strict SPF solution to work with segment routing traffic engineering (SR-TE).
- Redistribution of strict SPF sid is not supported.

# Information About OSPFv2 Segment Routing Strict SPF

Segment Routing (SR) architecture provides the provision to support multiple prefix-SID algorithms. Currently, it has defined two algorithms:

- **Algorithm 0** – This is a shortest path algorithm and it is supported by default.
- **Algorithm 1** – This is a strict shortest path algorithm. It mandates that the packets are forwarded according to SPF algorithm and instructs any router in the path to ignore any possible local policy overriding the SPF decision. The SID advertised with strict shortest path algorithm ensures that the path the packet is going to take is the expected path, and not the altered SPF path. You must configure strict SPF SID on each node that supports segment routing.

  The algorithm 1 is identical to algorithm 0, but it requires all the nodes along the path to honor the SPF routing decision. Local policy does not alter the forwarding decision. For example, a packet is not forwarded through locally engineered path.

## Why Strict SPF

In the case of link or node failure in the tunnel path, it is possible that MPLS traffic routed via the SR-TE tunnel is rerouted back to the tunnel head end from mid chain if the traffic is diverted to the repair path. If the head-end routes this MPLS traffic via the SR-TE tunnel again, then the same MPLS traffic may loop along the tunnel until TTL expires, even though there is an alternate IGP shortest path to the destination is available.

The strict SPF SID can prevent the looping of traffic through SR-TE tunnels. With strict SPF support, every router is configured to have both default SID, that is, SID0 and strict-SPF SID, that is, SID1. If the tunnel traffic is routed back to the head-end, it arrives at head end with strict-SPF SID as active label, which gets forwarded via the non-tunnel IGP shortest path (native path), thus breaking the looping along the SR-TE tunnel. Strict SPF prefix-SIDs are preferred over default prefix SIDs for SRTE tunnel when all nodes in the area/tunnel-path are Strict SPF capable.

## Strict-SPF Capability Advertisement

OSPF advertises the strict SPF capability in SR-Algorithm TLV of the Router Information (RI) opaque link state advertisements (LSA), when segment routing is enabled globally or on a specific area. OSPF includes both algorithm 0 (SPF) and algorithm 1 (Strict-SPF SID) in the SR-Algorithm TLV.

When received, OSPF parses the router information opaque LSA to find the SR Algorithm TLV. If the TLV is missing or algorithm 1 is not included in the TLV, OSPF ignores all strict-SPF SID advertisements from the advertisement router.

OSPF continues to support only single SRGB. The same SRGB is used for both regular SIDs and strict-SPF SIDs. Like regular SID, OSPF must not use out of SRGB range strict-SPF SIDs.

## Strict-SPF SID Advertisement in Extended Prefix LSA

OSPF advertises the strict SPF SID connected maps in prefix SID sub-TLV with algorithm set to 1 in OSPF extended prefix TLV of extended prefix opaque LSA. Both default SID and strict SPF SID for the same prefix are advertised in the same LSA. OSPF advertise separate explicit-NULL for regular and strict-SPF SIDs. Both the SIDs share same attach flag.

OSPF advertises the strict SPF SID mapping server entries in Prefix SID Sub-TLV with algorithm set to 1, in OSPF Extended Prefix Range TLV of Extended prefix opaque LSA. Both default SID and strict SPF SID may be advertised for the same prefix. If multiple SIDs of same algorithm are advertised for the same prefix, the receiving router uses the first encoded SID. OSPF advertise separate explicit-NULL for regular and strict-SPF SIDs. Both SIDs share same attach flag. The setting of attach flag in the regular SID takes over precedence if they differ.

If the SR-Algorithm TLV is missing or algorithm 1 is not included in the TLV, OSPF ignores all strict-SPF SID advertisements from the advertisement router. If multiple SIDs of same algorithm are received for the same prefix, the receiving router uses the first encoded SID.If the Explicit-NULL and Attach flags differ for the received SID0 & SID1 of a prefix, then the flags of SID0 takes over precedence.

## Interaction with SR-TE and Router Information Base

Like default SID, the strict-SPF SID also communicates with SR-TE only if SR and TE both are enabled for that area. There are three forms of communications that might happen with SR-TE related to strict-SPF SID:

- OSPF announces to SR-TE whether the area is strict-SPF capable or incapable. An area is strict SPF capable, if all the nodes in the area are strict spf capable and each node has at least one strict SPF SID configured.
- OSPF announces to SR-TE the strict-SPF SIDs for all prefixes and registered prefix paths.
- SR-TE prefers strict SPF SID for the label stack. OSPF receives tunnel list from SR-TE whenever there is a change to the list of auto-route announce tunnel list. For each tunnel, SR-TE indicates whether the tunnel is created using strict-SPF SIDs or default SIDs. OSPF runs full SPF whenever the updated tunnel list is received from SR-TE and replaces the RIB paths of prefixes reachable via the tunnel endpoint to the tunnel next hop.

Strict-SPF SIDs are not installed in the router information base (RIB). Only default SIDs get installed as the outgoing labels for the prefixes installed in the RIB. Both SR-TE tunnel types are installed in the RIB.

# Enabling and Disabling OSPFv2 Segment Routing Strict SPF

The strict SPF feature is enabled by default when segment-routing mpls is configured under OSPF and global mode. There is no separate CLI to enable or disable it.

# Configuring OSPFv2 Segment Routing Strict SPF SID

Perform the following steps to configure OSPFv2 segment routing strict SPF.

```
segment-routing mpls
 connected-prefix-sid-map
  address-family ipv4
   10.0.0.0/8 2
   172.16.0.0/8 3
  address-family ipv4 strict-spf
   10.0.0.0/8 22
   172.16.0.0/8 23
  exit-address-family
```

# Verifying OSPFv2 Segment Routing Strict SPF

Use the following commands to verify OSPFv2 segment routing strict SPF.

### Verifying OSPFv2 Segment Routing Strict SPF SID

```
Device#show ip ospf database opaque-area type ext-prefix

            OSPF Router with ID (10.0.0.4) (Process ID 10)

                Type-10 Opaque Area Link States (Area 0)

  LS age: 40
  Options: (No TOS-capability, DC)
  LS Type: Opaque Area Link
  Link State ID: 7.0.0.3
  Opaque Type: 7 (Extended Prefix)
  Opaque ID: 3
  Advertising Router: 10.0.0.2
  LS Seq Number: 80000003
  Checksum: 0xFB42
  Length: 56

    TLV Type: Extended Prefix
    Length: 32
      Prefix    : 10.0.0.6/32
      AF        : 0
      Route-type: Intra
      Flags     : N-bit

      Sub-TLV Type: Prefix SID
      Length: 8
        Flags : None
        MTID  : 0
        Algo  : SPF
        SID   : 100

      Sub-TLV Type: Prefix SID
      Length: 8
        Flags : None
        MTID  : 0
```

```
        Algo  : Strict SPF
        SID   : 101

Device#show ip ospf segment-routing sid-database


            OSPF Router with ID (10.0.0.4) (Process ID 10)

OSPF Segment Routing SIDs

Codes: L - local, N - label not programmed,
       M - mapping-server

SID             Prefix              Adv-Rtr-Id       Area-Id  Type      Algo
--------------  ------------------  ---------------  -------  --------  ----
2               10.0.0.2/32          10.0.0.2          0        Intra     0
4       (L)     10.0.0.4/32          10.0.0.4          0        Intra     0
7               10.0.0.7/32          10.0.0.5          0        Intra     0
9               10.0.0.8/32          10.0.0.2          0        Intra     0
20              2.0.2.20/32         2.2.2.2          0        Intra     0
21              22.0.22.21/32       2.2.2.2          0        Intra     1
22      (M)     2.0.2.22/32                                   Unknown   0
29      (M)     22.0.22.29/32                                 Unknown   1
33              33.0.33.33/32       3.3.3.3          0        Intra     1
38      (M)     3.0.3.38/32                                   Unknown   0
39      (M)     33.0.33.39/32                                 Unknown   1
77              77.77.77.77/32      5.5.5.5          0        Inter     0
92      (M)     2.1.2.92/32                                   Unknown   0
99              99.99.99.99/32      9.9.9.9          0        Intra     0
100             2.0.2.100/32        2.2.2.2          0        Intra     0
101             2.0.2.100/32        2.2.2.2          0        Intra     1
120             3.3.3.120/32        3.3.3.3          0        Intra     0
121             3.3.3.120/32        3.3.3.3          0        Intra     1

Device#show ip ospf segment-routing mapping-server

            OSPF Router with ID (10.0.0.4) (Process ID 10)

Advertise local: Enabled
Receive remote: Enabled

Flags: i - sent to mapping-server, u - unreachable,
       s - self-originated

2.0.2.22/32 (R), range size 1
   Adv-rtr        Area          LSID          SID    Type    Algo
i  2.2.2.2        0             7.0.0.4       22     Intra   0
s  4.4.4.4        24            7.0.0.1       22     Inter   0

2.1.2.92/32 (R), range size 1
   Adv-rtr        Area          LSID          SID    Type    Algo
i  2.2.2.2        0             7.0.0.5       92     Intra   0
s  4.4.4.4        24            7.0.0.2       92     Inter   0

3.0.3.38/32 (R), range size 1
   Adv-rtr        Area          LSID          SID    Type    Algo
i  3.3.3.3        0             7.0.0.2       38     Intra   0
s  4.4.4.4        24            7.0.0.3       38     Inter   0

3.3.3.48/32 (R), range size 1
   Adv-rtr        Area          LSID          SID    Type    Algo
i  3.3.3.3        0             7.0.0.3       48     Intra   0
s  4.4.4.4        24            7.0.0.4       48     Inter   0
```

```
22.0.22.29/32 (R), range size 1
   Adv-rtr        Area          LSID          SID    Type    Algo
i  2.2.2.2        0             7.0.0.6       29     Intra   1
s  4.4.4.4        24            7.0.0.5       29     Inter   1

22.1.22.99/32 (R), range size 1
   Adv-rtr        Area          LSID          SID    Type    Algo
i  2.2.2.2        0             7.0.0.7       99     Intra   1
s  4.4.4.4        24            7.0.0.6       99     Inter   1

33.0.33.39/32 (R), range size 1
   Adv-rtr        Area          LSID          SID    Type    Algo
i  3.3.3.3        0             7.0.0.4       39     Intra   1
s  4.4.4.4        24            7.0.0.7       39     Inter   1

33.3.33.49/32 (R), range size 1
   Adv-rtr        Area          LSID          SID    Type    Algo
i  3.3.3.3        0             7.0.0.5       49     Intra   1
s  4.4.4.4        24            7.0.0.8       49     Inter   1
```

```
Device#show ip ospf segment-routing local-prefix
            OSPF Router with ID (10.0.0.7) (Process ID 10)
Area 0:
 Prefix:            Sid:   Index:          Type:      Algo: Source:
 2.2.2.2/32         2      0.0.0.0         Intra      0     Loopback0
                         22     0.0.0.0         Intra      1     Loopback0
 23.23.23.4/32      233    0.0.0.1         Intra      1     Loopback3
```

## Verifying OSPFv2 Segment Routing Strict SPF Capability

```
Device#show ip ospf database opaque-area type router-information self

            OSPF Router with ID (10.0.0.4) (Process ID 10)

                Type-10 Opaque Area Link States (Area 0)

  LS age: 1692
  Options: (No TOS-capability, DC)
  LS Type: Opaque Area Link
  Link State ID: 4.0.0.0
  Opaque Type: 4 (Router Information)
  Opaque ID: 0
  Advertising Router: 4.4.4.4
  LS Seq Number: 80000002
  Checksum: 0x72B
  Length: 60

    TLV Type: Router Information
    Length: 4
    Capabilities:
      Graceful Restart Helper
      Stub Router Support
      Traffic Engineering Support

    TLV Type: Segment Routing Algorithm
    Length: 2
      Algorithm: SPF
      Algorithm: Strict SPF

    TLV Type: Segment Routing Range
    Length: 12
```

```
      Range Size: 8000

      Sub-TLV Type: SID/Label
      Length: 3
        Label: 16000

    TLV Type: Segment Routing Node MSD
    Length: 2
      Sub-type: Node Max Sid Depth, Value: 10
```

### Verifying Strict SPF Labels Used in OSPF Local RIB Database

```
Device#show ip ospf rib 10.0.0.8

            OSPF Router with ID (10.0.0.6) (Process ID 10)


                Base Topology (MTID 0)

OSPF local RIB
Codes: * - Best, > - Installed in global RIB
LSA: type/LSID/originator

*>  2.0.2.100/32, Intra, cost 21, area 0
     SPF Instance 28, age 00:01:19
      contributing LSA: 10/7.0.0.3/2.2.2.2 (area 0)
     SID: 100, Properties: Sid, LblRegd, SidIndex, N-Flag, TeAnn
     Strict SPF SID: 101, Properties: Force, Sid, LblRegd, SidIndex, N-Flag
     Flags: RIB, HiPrio
      via 3.6.0.3, Ethernet0/1, label 16100, strict label 16101
       Flags: RIB
       LSA: 1/2.2.2.2/2.2.2.2
      PostConvrg repair path via 5.6.0.5, Ethernet0/3, label 16100, strict label 16100,
cost 31
       Flags: RIB, Repair, PostConvrg, IntfDj, BcastDj
       LSA: 1/2.2.2.2/2.2.2.2
```

### Verifying Strict SPF TILFA Tunnels

```
Device#show ip ospf fast-reroute ti-lfa tunnels internal

            OSPF Router with ID (10.0.0.2) (Process ID 10)

                        Area with ID (0)

                    Base Topology (MTID 0)


TI-LFA Release Node Tree:

TI-LFA Release Node 4.4.4.4 via 1.2.0.1 Ethernet0/0, instance 12, metric 20
  Interface MPLS-SR-Tunnel2
    Tunnel type: MPLS-SR (strict spf)
    Tailend router ID: 4.4.4.4
    Termination IP address: 4.4.4.4
    Outgoing interface: Ethernet0/0
    First hop gateway: 1.2.0.1
    instance 12, refcount 1
      rn-1: rtrid 4.4.4.4, addr 4.4.4.4, strict node-sid label 16044

TI-LFA Release Node 4.4.4.4 via 2.3.0.3 Ethernet0/1, instance 12, metric 20
  Interface MPLS-SR-Tunnel1
```

```
    Tunnel type: MPLS-SR (strict spf)
    Tailend router ID: 4.4.4.4
    Termination IP address: 4.4.4.4
    Outgoing interface: Ethernet0/1
    First hop gateway: 2.3.0.3
    instance 12, refcount 1
      rn-1: rtrid 4.4.4.4, addr 4.4.4.4, strict node-sid label 16044


TI-LFA Node Tree:

TI-LFA Node 1.1.1.1 via 1.2.0.1 Ethernet0/0, abr, instance 12, rspt dist 0
  not-in-ext-p-space, in-q-space, interesting node 1
  Link Protect strict Path-1: via 2.3.0.3 Et0/1, parent 1/4.4.4.4, metric:30, rls-pt:4.4.4.4
 at dist:20
  repair:y, rn-cnt:1, first-q:4.4.4.4, rtp-flags:Repair, PostConvrg, IntfDj
    rn-1: rtrid 4.4.4.4, addr 4.4.4.4, strict node-sid label 16044
    Protected by: MPLS-SR-Tunnel1, tailend 4.4.4.4, rls node 4.4.4.4
    instance 12, metric 20, refcount 1


TI-LFA Node 3.3.3.3 via 2.3.0.3 Ethernet0/1, instance 12, rspt dist 0
  not-in-ext-p-space, in-q-space, interesting node 1
  Link Protect strict Path-1: via 1.2.0.1 Et0/0, parent 1/4.4.4.4, metric:30, rls-pt:4.4.4.4
 at dist:20
  repair:y, rn-cnt:1, first-q:4.4.4.4, rtp-flags:Repair, PostConvrg, IntfDj
    rn-1: rtrid 4.4.4.4, addr 4.4.4.4, strict node-sid label 16044
    Protected by: MPLS-SR-Tunnel2, tailend 4.4.4.4, rls node 4.4.4.4
    instance 12, metric 20, refcount 1

TI-LFA Node 4.4.4.4 via 1.2.0.1 Ethernet0/0, abr, instance 12, rspt dist 10
  in-ext-p-space, in-q-space, interesting node 1
  Link Protect strict Path-1: via 2.3.0.3 Et0/1, parent 1/3.3.3.3, metric:20, rls-pt:3.3.3.3
 at dist:10
  repair:y, rn-cnt:0, first-q:4.4.4.4, rtp-flags:Repair, PostConvrg, IntfDj, PrimPath
    Protected by: directly connected TI-LFA

TI-LFA Node 4.4.4.4 via 2.3.0.3 Ethernet0/1, abr, instance 12, rspt dist 10
  in-ext-p-space, in-q-space, interesting node 1
  Link Protect strict Path-1: via 1.2.0.1 Et0/0, parent 1/1.1.1.1, metric:20, rls-pt:1.1.1.1
 at dist:10
  repair:y, rn-cnt:0, first-q:4.4.4.4, rtp-flags:Repair, PostConvrg, IntfDj, PrimPath
    Protected by: directly connected TI-LFA


TI-LFA Protected neighbors:

  Neighbor 1.2.0.1 Ethernet0/0, ID 1.1.1.1, Dist 10, instance 12
    TI-LFA Required, TI-LFA Computed, RLFA not Required
    TI-LFA protection Required: link

  Neighbor 2.3.0.3 Ethernet0/1, ID 3.3.3.3, Dist 10, instance 12
    TI-LFA Required, TI-LFA Computed, RLFA not Required
    TI-LFA protection Required: link
```

## Verifying Strict SPF SR-TE Tunnels

```
Device#show mpls traffic-eng segment-routing ospf summary
IGP Area[1]:  ospf 10  area 0, Strict SPF Enabled:
Nodes:
IGP Id: 1.1.1.20, MPLS TE Id: 1.1.1.1, OSPF area 0
```

```
   2 links with segment-routing adjacency SID
IGP Id: 2.0.0.0, MPLS TE Id: 2.2.2.2, OSPF area 0
   2 links with segment-routing adjacency SID
IGP Id: 3.0.0.0, MPLS TE Id: 3.3.3.3, OSPF area 0
   3 links with segment-routing adjacency SID
IGP Id: 4.4.4.4, MPLS TE Id: 4.4.4.4, OSPF area 0
   3 links with segment-routing adjacency SID
IGP Id: 5.0.0.0, MPLS TE Id: 5.5.5.5, OSPF area 0
   2 links with segment-routing adjacency SID
Prefixes:
1.1.1.1/32, SID index: 1, Strict SID index: 11
1.2.0.2/32
2.2.2.2/32, SID index: 2, Strict SID index: 22
2.2.2.22/32, SID index: 222, Strict SID index: 2222
3.3.3.3/32, SID index: 3, Strict SID index: 34
3.3.3.33/32, SID index: 333, Strict SID index: 1333
4.4.4.4/32, SID index: 4, Strict SID index: 444
5.5.5.5/32, SID index: 5, Strict SID index: 555
6.6.6.6/32, SID index: 6
7.7.7.7/32, SID index: 7
Total:
  Node Count         : 5
  Adjacency-SID Count: 17
  Prefix-SID Count   : 10
Grand Total:
  Node Count         : 5
  Adjacency-SID Count: 17
  Prefix-SID Count   : 10
  IGP Areas Count    : 1
```

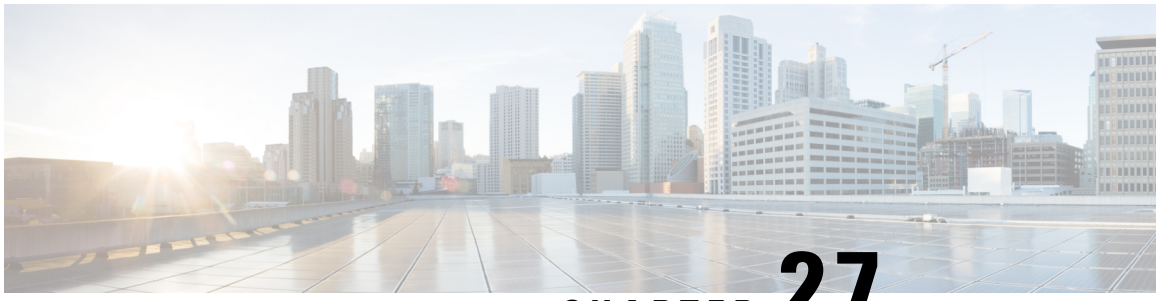
### Verifying Protected adj-SIDs Using Strict SPF Repair Path

```
Device#sh ip ospf segment-routing protected-adjacencies detail

            OSPF Router with ID (10.0.0.0) (Process ID 10)

                        Area with ID (0)

Nbr id 10.0.0.1, via 10.0.0.2 on Ethernet0/1, Label 26
  Primary path: via 10.0.0.2 on Et0/1, out-label 3
  Repair path: via 10.0.0.3 on Et0/2, out-label 13222, cost 31, labels 0
  Nbr Prefix 10.0.0.4, Strict
Nbr id 10.0.0.5, via 10.0.0.3 on Ethernet0/2, Label 25
  Primary path: via 10.0.0.3 on Et0/2, out-label 3
  Repair path: via 10.0.0.2 on Et0/1, out-label 12333, cost 21, labels 0
  Nbr Prefix 10.0.0.5, Strict
```

### Verifying Segment Routing Global Block

```
Device#show ip ospf segment-routing global-block

            OSPF Router with ID (10.0.0.0) (Process ID 10)

OSPF Segment Routing Global Blocks in Area 0

  Router ID:      SR Capable: SR Algorithm: SRGB Base: SRGB Range:  SID/Label:

 *10.0.0.0         Yes         SPF,StrictSPF 16000      8000         Label
  10.0.0.1         Yes         SPF,StrictSPF 16000      8000         Label
  10.0.0.2         Yes         SPF,StrictSPF 16000      8000         Label
  10.0.0.3         Yes         SPF           16000      8000         Label
  10.0.0.4         Yes         SPF,StrictSPF 16000      8000         Label
```

```
  10.0.0.5         No
  10.0.0.6         Yes         SPF           16000      8000         Label
Device#
```



CHAPTER 27

# Segment Routing OSPFv2 Microloop Avoidance

The feature enables link-state routing protocols such as IS-IS and OSPF to prevent or avoid microloops during network convergence after a topology undergoes any change.


- Feature Information for Segment Routing OSPFv2 Microloop Avoidance, on page 247
- Information About Segment Routing OSPFv2 Microloop Avoidance, on page 248
- Prerequisites for Segment Routing OSPFv2 Microloop Avoidance, on page 251
- Restrictions for Segment Routing OSPFv2 Microloop Avoidance, on page 252
- Configuring Segment Routing OSPFv2 Microloop Avoidance, on page 252
- Verifying Segment Routing OSPFv2 Microloop Avoidance, on page 252


# Feature Information for Segment Routing OSPFv2 Microloop Avoidance

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

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

***Table 27: Feature Information for Segment Routing OSPFv2 Microloop Avoidance***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Segment Routing OSPFv2 Microloop Avoidance | Cisco IOS XE Amsterdam 17.3.2 | The Segment Routing microloop avoidance feature enables link-state routing protocols such as IS-IS and OSPF to prevent or avoid microloops during network convergence after a topology change.<br><br>The following commands was introduced/modified by this feature: **microloop avoidance segment-routing**. |

# Information About Segment Routing OSPFv2 Microloop Avoidance

Microloops are brief packet loops that occur in the network following a topology change (link down, link up, or metric change events). Microloops are caused by the non-simultaneous convergence of different nodes in the network. If nodes converge and send traffic to a neighbor node that has not converged, traffic may be looped between these two nodes, resulting in packet loss, jitter, and out-of-order packets.

If segment routing microloop avoidance feature detects a topology change, it creates a loop-free path to the destination using a list of segments.

## Microloops

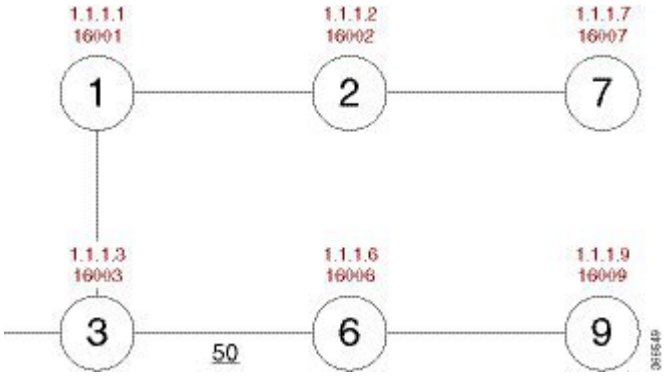
When changes occur in a network topology because of the failure or restoration of a link or a network device, IP Fast Reroute enables rapid network convergence by moving traffic to precomputed backup paths until regular convergence mechanisms move traffic to a newly computed best path, which is also known as a post-convergence path. This network convergence may cause short microloops between two directly or indirectly connected devices in the topology. Microloops are caused when different nodes in the network calculate alternate paths at different times and independently of each other. For instance, if a node converges and sends traffic to a neighbor node, which has not converged yet, traffic may loop between the two nodes.

Microloops may or may not result in traffic loss. If the duration of a microloop is short, that is the network converges quickly, packets may loop for a short duration before their time-to-live (TTL) expires. Eventually, the packets will get forwarded to the destination. If the duration of the microloop is long, that is one of the routers in the network is slow to converge, packets may expire their TTL or the packet rate may exceed the bandwidth, or the packets might be out of order, and packets may dropped.

Microloops that are formed between a failed device and its neighbors are called local uloops, whereas microloops that are formed between devices that are multiple hops away are called remote uloops. Local uloops are usually seen in networks where local loop-free alternate (LFA) path is not available. In such networks, remote LFAs provide backup paths for the network.

The information discussed above can be illustrated with the help of an example topology.

***Figure 29: Microloop Example Topology***



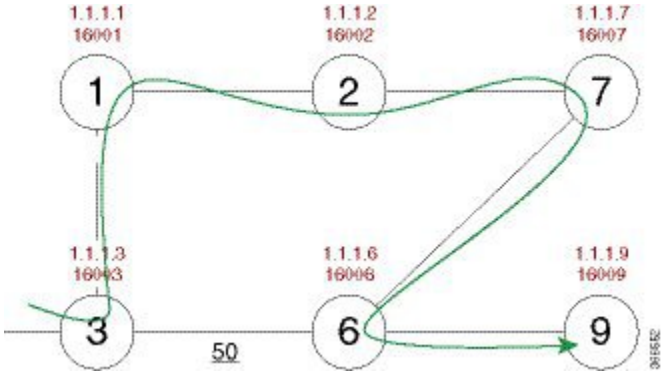

The assumptions in this example are as follows:

- The default metrics is 10 for each link except for the link between Node 3 and Node 6, which has a metric of 50. The order of convergence with SPF backoff delays on each node is as follows:
  - Node 3—50 milliseconds
  - Node 1—500 milliseconds
  - Node 2—1 second
  - Node 7—1.5 seconds

A packet sent from Node 3 to Node 9, the destination, traverses via Node 6.

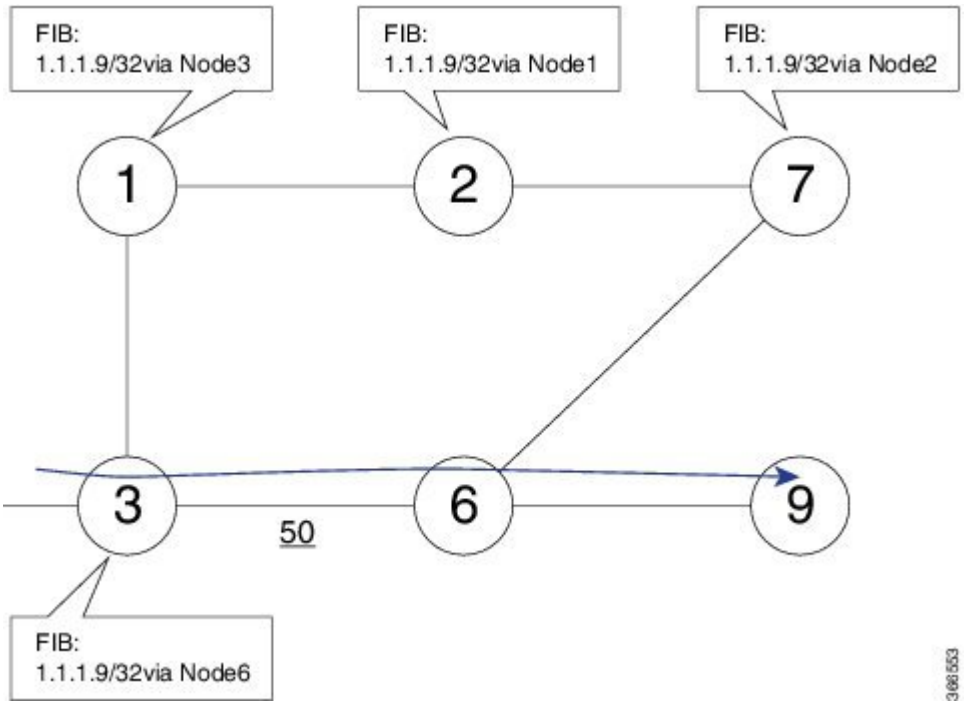
If a link is established between Node 6 and Node 7, the shortest path for a packet from Node 3 to Node 9 would be Node 1, Node 2, Node 7, and Node 6 before the packet reaches the destination, Node 9.

***Figure 30: Microloop Example Topology—Shortest Path***



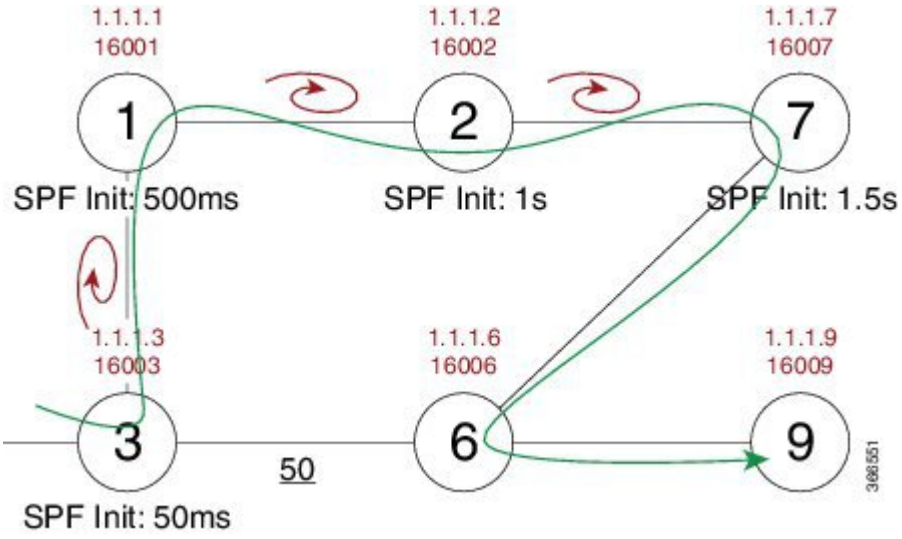

The following figure shows the Forwarding Information Base (FIB) table in each node before the link between Node 6 and Node 7 is established. The FIB entry contains the prefix of the destination node (Node 9) and the next hop.

*Figure 31: Microloop Example Topology—FIB Entry*




When the link between Node 6 and Node 7 comes up, microloops occur for the links based on the order of convergence of each node. In this example, Node 3 converges first with Node 1 resulting in a microloop between Node 3 and Node 1. Then, Node 1 converges next resulting in a microloop between Node 1 and Node 2. Next, Node 2 converges next resulting in a microloop between Node 2 and Node 7. Finally, Node 7 converges resolving the microloop and the packet reaches the destination Node 9, as shown in the following figure.

*Figure 32: Microloop Example Topology—Microloops*

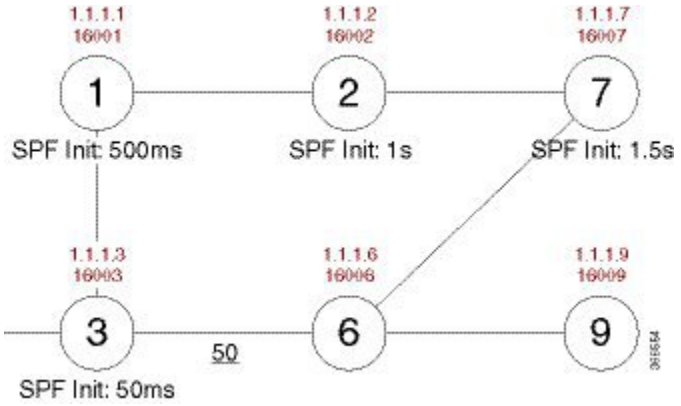



Adding the SPF convergence delay, microloop results in a loss of connectivity for 1.5 seconds, which is the convergence duration specified for node 7.

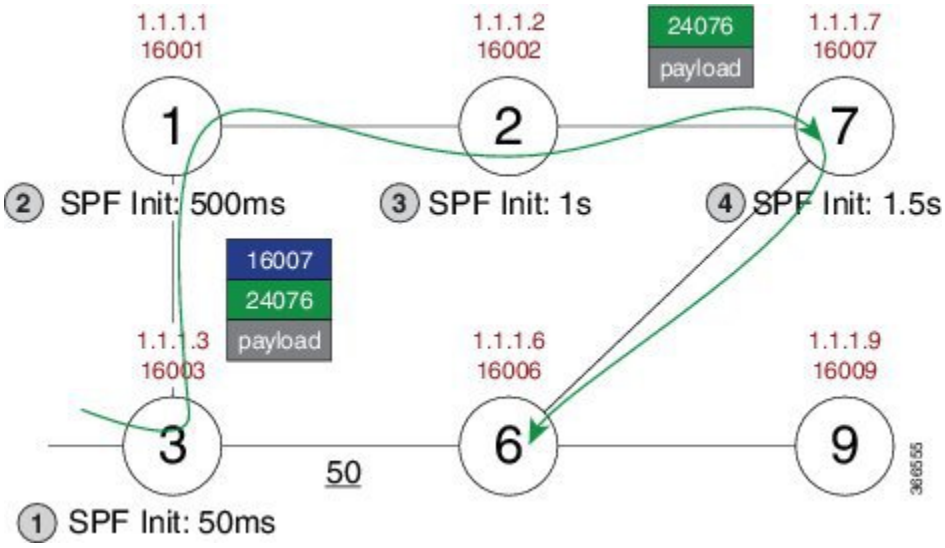
# Preventing Microloops using Segment Routing

This section explains how segment routing prevents microloops using an example. Node 3 in the example is enabled with the **microloop avoidance segment-routing** command.

***Figure 33: Microloop Example Topology—Segment Routing***




Instead of updating the FIB table, Node 3 builds a dynamic loop-free path to the destination (Node 9) using a list of segments IDs, which include the prefix segment ID (SID) of Node 7, which is 16007, and the adjacency segment ID (SID) of Node 6, which is 24076.




So the packet from Node 3 reaches its destination Node 9 without the risk of microloop until the network converges. Finally, Node 3 updates the FIB with the new path.

# PrerequisitesforSegmentRoutingOSPFv2MicroloopAvoidance

Before configuring SR microloop avoidance, ensure that the segment routing is globally configured in the OSPF router mode.

```
router ospf process
segment-routing mpls
```

# Restrictions for Segment Routing OSPFv2 Microloop Avoidance

- Segment Routing OSPFv2 microloop avoidance does not support Multi Topology Routing (MTR). It supports only MTID 0.
- A list of segment IDs along the post convergence path is used only if the nodes in the the list are SR capable and have atleast one node SID. Otherwise, OSPF installs the post convergence path immediately.
- SR microloop avoidance is used for link up, link down, and link metric change events of point-to-point interfaces and broadcast interfaces with two neighbors only.
- SR microloop avoidance can be used only for one topology change. When multiple topology changes occur, OSPF installs the post convergence path immediately.

# Configuring Segment Routing OSPFv2 Microloop Avoidance

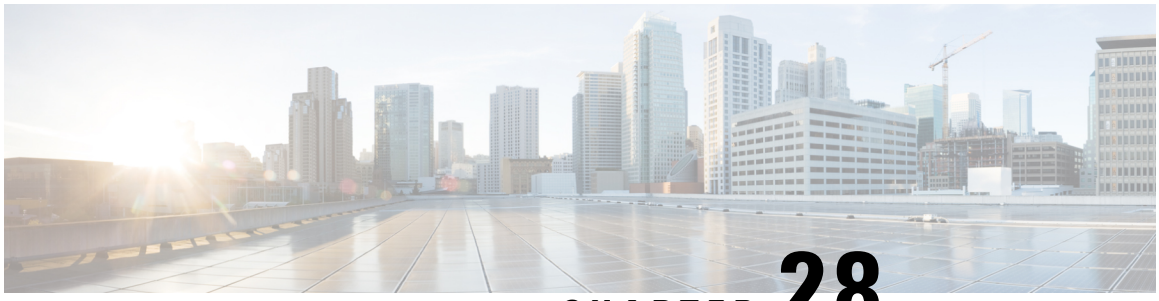
Enables segment routing microloop avoidance for all the prefixes.

```
router ospf
 microloop avoidance segment-routing
 microloop avoidance rib-update-delay delay-time
```

The **microloop avoidance rib-update-delay** *delay-time* command is used to configure the delay in milliseconds for a node to wait before updating the node's forwarding table and stops using the microloop avoidance. The default value for the RIB delay is 5000 milliseconds.

# Verifying Segment Routing OSPFv2 Microloop Avoidance

Use the **show ip ospf segment-routing microloop avoidance** command to check if SR microloop avoidance is enabled or not.



CHAPTER 28

# Performance Measurement for Traffic Engineering

Metrics such as packet loss, delay, delay variation (jitter) and bandwidth utilization help you evaluate the performance of your network. You can use these metrics as input for Traffic Engineering (TE) and direct the flow of traffic through the network to conform to Service Level Agreements (SLAs). With this feature, you can configure the measurement and advertisement of link delay metrics for TE.


- Feature Information for Performance Measurement for Traffic Engineering, on page 253
- Information about Performance Metrics for Traffic Engineering, on page 254
- How to Configure Performance Measurement for Traffic Engineering, on page 258
- Additional References, on page 263


## Feature Information for Performance Measurement for Traffic Engineering

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

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

***Table 28: Feature Information for Performance Measurement for Traffic Engineering***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Link Delay Measurement | Cisco IOS XE Amsterdam 17.3.2 | Metrics such as packet loss, delay, delay variation (jitter) and bandwidth utilization help you evaluate the performance of your network. You can use these metrics as input for Traffic Engineering (TE) and direct the flow of traffic through the network to conform to Service Level Agreements (SLAs). With this feature, you can configure the measurement and advertisement of link delay metrics for TE. |

# Information about Performance Metrics for Traffic Engineering

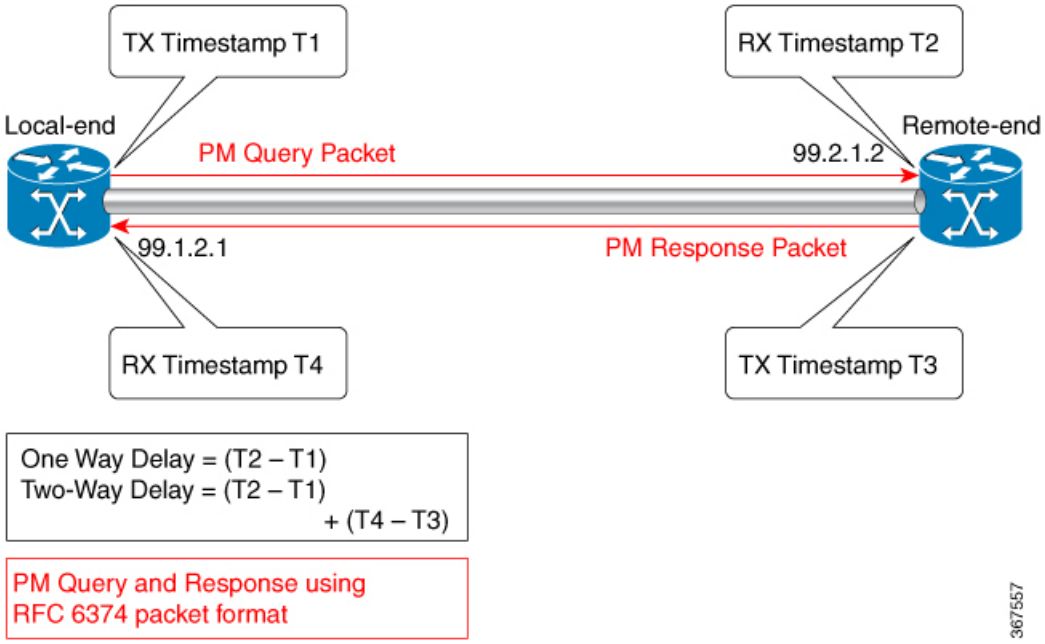
## Overview of Link Delay Measurement

Link delay is measured using PM Query packets in the format defined in RFC 6374. To support the packet format, the remote line card must be MPLS capable.

**Note** Only two-way link delay measurement is supported.

For link delay measurement, an MPLS multicast MAC address is used to send delay measurement probe packets to next-hops. You need not configure next-hop addresses for the links. The remote side line card must support the MPLS multicast MAC address.

The following figure shows the measurement of link delay using the PM Query and Response packets.



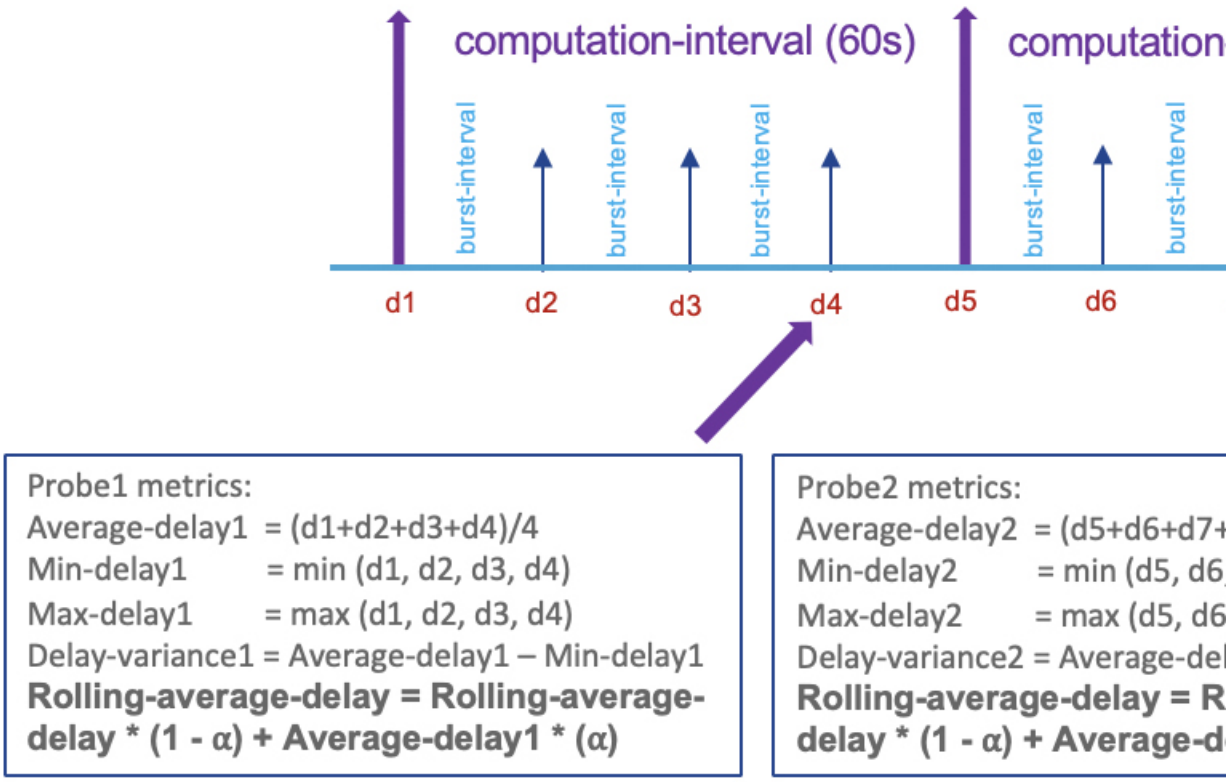

1. The local-end router sends a burst of PM Query packets to the remote-end router at the configured interval. The packets are timestamped (T1) before they are sent.
2. At the remote-end router, packets are timestamped (T2) when they are received.
3. The remote-end router sends the PM packets containing the timestamps (T1 and T2) back to the local-end router. The packets are timestamped (T3) before they are sent.
4. At the local-end router, packets are timestamped (T4) when they are received.
5. At the local-end router, two-way link delay is measured using the timestamps of the PM packets.

## Link Delay Metrics for a Computation Interval

The local-end router sends a configured count of PM query packets to the remote-end router at configured burst intervals. The local-end router measures two-way link delay for each burst of PM Query packets that it sends to the remote-end router and receives back with timestamps.

During each configured probe or computation interval, multiple bursts of PM packets are sent and link delay is measured. Minimum, maximum, and average link delay, and delay variance are calculated for the interval. These metrics are calculated using the link delay that is measured for the bursts sent during the interval.

The following figure illustrates the calculation of delay metrics for a computation interval. In this example, the computation interval is 60 seconds and the burst interval is 15 seconds.




## Link Delay Metrics for Advertisement

You can configure the computation and advertisement of delay metrics in a periodic manner, an accelerated manner, or both. The advertisement of link delay metrics is supported with the ISIS, OSPF, and BGP-LS protocols. No additional configuration is required to flood link delay metrics through ISIS, OSPF, and BGP-LS protocols.

### Periodic Advertisement

Periodic advertisement is enabled by default. A periodic advertisement interval consists of one or more computation or probe intervals. Link delay metrics are computed at the end of each computation interval. In a periodic advertisement interval, after the last computation interval, the minimum delay computed for a link is compared with the value advertised previously. If the variation in values is beyond configured limits, all

the delay metrics for the link are advertised. If the variation in values is within configured limits, the delay metrics for the link are not advertised.

- Suppose a periodic advertisement interval consists of N computation intervals, at the end of a computation interval *i*, the following metrics are computed:
  - Rolling average delay

    Rolling average delay = rolling-average-delay(i-1) * 0.5 + average-delay(i) * 0.5
  - Minimum delay

    Minimum delay = min[min-delay(1),..., min-delay(i-1), min-delay(i)]
  - Maximum delay

    Maximum delay = max[max-delay(1),…, max-delay(i-1), max-delay(i)]
  - Delay variance

    Delay variance = average[delay-variance(1),…, delay-variance(i-1), delay-variance(i)]
- After the last computation interval in the periodic advertisement interval, the minimum delay for a link is compared with the value advertised after the previous interval.
  - Case 1: change between the two values is beyond the configured threshold and minimum-change. In this case, all the delay metrics computed for the link after the recent periodic advertisement interval are advertised.
  - Case 2: change between the two values is within the configured threshold and minimum-change. In this case, the delay metrics are not advertised.

#### Accelerated Advertisement

By default, accelerated advertisement is disabled. When you enable accelerated advertisement, the minimum link delay that is computed for a link after a computation interval is compared with the value previously advertised. If the variation in values is beyond configured limits, all the delay metrics for the link are advertised. If the variation in values is within configured limits, the delay metrics for the link aren't advertised.

When link delay metrics are advertised in an accelerated manner, the periodic advertisement interval is reset. This reset ensures the configured interval of time between the recent advertisement and the next periodic assessment.

#### Link Delay Metrics when the Link State Changes

When a link enters the DOWN state, link delay metrics are advertised with the highest value. The minimum, maximum, and average link delay, and delay variance are advertised with a value of 16.7 seconds (0xFFFFFF). With the highest metric values advertised, routing and SR-TE path computation don't use stale metric values when the link enters the UP state.

## Global Link Delay Profile

You can configure a global profile for the measurement of link delay metrics. The profile defines parameters that control the computation and advertisement of link delay metrics and replaces the default configuration. Being global, the profile applies to link delay measurement on all interfaces.

You can configure the following parameter as part of the global profile:

*Table 29: Global Link Delay Profile Parameters*

| Aspect | Parameter | Description |
|---|---|---|
| probe | interval | The default probe or computation interval is 30 seconds. The range is 30–3600 seconds. |
| | protocol | Protocol used to send probes. The default and the only supported protocol is pm-mpls: link delay measurement based on RFC 6374 with MPLS encapsulation. |
| burst | count | The default value is 10 and range is 1–30. |
| | interval | The default value is 3000 milliseconds and the range is 30–15000 milliseconds. |
| periodic advertisement | interval | The default value is 120 seconds and the interval range is 30–3600 seconds. |
| | threshold | The default value of periodic advertisement threshold is 10 percent. |
| | minimum-change | The default value is 1000 microseconds and the range is 0–10000 microseconds. |
| | disabled | Periodic advertisement is enabled by default. |
| accelerated advertisement | threshold | The default value is 20 percent and the range is 0–100 percent. |
| | minimum-change | The default value is 1000 microseconds and the range is 1–100000 microseconds. |

# Benefits of Link Delay Measurement

You can use link delay metrics such as average, minimum, and maximum delay, and delay variance to determine network latency. Using link delay metrics, you can troubleshoot latency issues or apply Traffic Engineering (TE) solutions to meet Service Level Agreements (SLAs). For example, you could

- configure SR Policies that have acceptable delay
- steer traffic through alternative SR Policies when the delay performance of the serving SR Policies deteriorates beyond acceptable limits.

# Restrictions for Link Delay Measurement

### Restrictions in IOS XE Release 17.1.x

- Measurement of only two-way link delay is supported.
- PM link delay measurement is based on RFC 6374 and the PM packets use MPLS/GAL encapsulation.
- Only minimum-delay value is used for threshold checks.
- You cannot configure the packet size and TOS/DSCP/EXP of link-delay probe protocol packets.
- Link delay values that exceed two seconds are discarded.

# How to Configure Performance Measurement for Traffic Engineering

## Configuring Global Link Delay Profile

Configure the parameters of the global link delay profile by entering the interface delay profile mode:

```
performance-measurement
    delay-profile
        interfaces    ---> Global default profile for link delay measurement
            probe
                interval <seconds>  (range:30-3600 seconds; default:30 seconds)
                burst
                    count <num-of-packets>  (range:1-30; default: 10)
                    interval <milliseconds>  (range:30-15000 milliseconds; default:3000 milliseconds)
                protocol
                   pm-mpls                   SR Policy delay measurement using RFC6374 with MPLS encapsulation

            advertisement
                periodic                        (default: enabled)
                    disabled
                    interval <seconds>             (range:30-3600 seconds; default:120 seconds)
                    threshold <percentage>         (range:0-100%; default:10%)
                    minimum-change <microseconds>  (range:0-100000 microseconds; default: 1000 microseconds)
                accelerated                (default: disabled)
                    threshold <percentage>         (range:0-100%; default: 20%)
                    minimum-change <microseconds>  (range:0-100000 microseconds; default: 1000 microseconds)
```

## Configuring Link Delay Measurement for an Interface

### Enabling Link Delay Measurement for an Interface

Enable delay-measurement for an interface as follows:

```
performance-measurement
    interface <interface-name>
        delay-measurement
```

### Disabling Link Delay Measurement for an Interface

Disable delay-measurement for an interface as follows:

```
performance-measurement
    interface <interface-name>
        no delay-measurement
```

### Configuring a Link Delay for an Interface

Set a link delay for an interface as follows:

```
performance-measurement
    interface <interface-name>
        delay-measurement
            advertise-delay <microseconds>    (range: 0-16777215 microseconds)
```

When the advertise-delay is set for an interface,

- the minimum, maximum, and average delays for the associated link are set to the advertise-delay value
- the delay variance for the link is set to zero
- the link delay metrics are immediately advertised.

During the computation interval, PM query and response packets are exchanged and link delay metrics are computed. These metrics are stored in the history buffer and can be accessed using the command **show performance-measurement history interfaces** [**name** *interface-name*] [**adv** | **aggr** | **probe**]. However, when advertise-delay is configured, threshold checks are not performed. Therefore, the computed metrics are not advertised.

Remove the set link delay for an interface as follows:

```
performance-measurement
    interface <interface-name>
        delay-measurement
            no advertise-delay <microseconds>    (range: 0-16777215 microseconds)
```

When the set link delay is removed for an interface,

- delay metrics are unpublished by removing TLVs from the IGP,
- at the end of the subsequent advertisement interval, threshold checks are performed. Based on the threshold checks, link delay metrics are advertised if necessary.

# Enabling Monitoring Mode

In the Monitoring Mode, the computed delay metrics are stored in the history buffer. However, the metrics are not advertised by an IGP or BGP-LS. You can display the metrics in the history buffer using the **show performance-measurement history interfaces** [**name** *interface-name*] [**adv** | **aggr** | **probe**] command.

To enable Monitoring Mode, disable both periodic and accelerated advertisement of link delay metrics.

**Note** Accelerated advertisement is disabled by default.

Disable periodic advertisement as follows:

```
performance-measurement
    delay-profile
        interfaces    ---> Global default profile for link delay measurement
            advertisement
                periodic                          (default: enabled)
                    disabled
```

With Monitoring Mode enabled,

- link delay metrics are not published through Interface Manager attributes in the system.
- link delay metrics are not flooded in the network by IGPs or advertised by BGP-LS.

# Verifying Link Delay Configuration

Use the **show performance-measurement summary** [**detail**] command to view the link delay configuration.

### Example

```
router#show performance-measurement summary
Total interfaces                    : 2
Maximum PPS                         : 100 pkts/sec

Interface Delay-Measurement:
  Total sessions                    : 2
  Profile configuration:
    Measurement Type                : Two-Way
    Probe interval                  : 30 seconds
    Burst interval                  : 3000 mSec
    Burst count                     : 10 packets
    Protocol                        : MPLS RFC6374
    HW Timestamp Supported          : Yes
    Periodic advertisement          : Enabled
      Interval                      : 120 (effective: 120) sec
      Threshold                     : 10%
      Minimum-Change                : 1000 uSec
    Advertisement accelerated       : Disabled
    Threshold crossing check        : Minimum-delay
  Counters:
    Packets:
      Total sent                    : 289588
      Total received                : 289588
    Errors:
      Total sent errors             : 23
      Total received errors         : 21
        .
        .
        .
```

# Viewing Link Delay Information for an Interface

Use the **show performance-measurement interfaces** [**name** *interface-name*] [**detail**] command to view information about the link delay measurement for an interface.

### Example

```
router#show performance-measurement interfaces name gigabitEthernet 0/0/7 detail
Interface Name: GigabitEthernet0/0/7 (ifh: 0xF)
  Delay-Measurement           : Enabled
  Local IPV4 Address          : 100.0.1.1
  Local IPV6 Address          : ::
```

```
State                       : Up

Delay Measurement session:
  Session ID                : 1

  Last advertisement:
    Advertised at: 13:53:11  28 2019 (434548 seconds ago)
    Advertised reason: Periodic timer, min delay threshold crossed
    Advertised delays (uSec): avg: 4011, min: 4033, max: 4050, variance: 4

  Next advertisement:
    Check scheduled in 2 more probes (roughly every 120 seconds)
    Aggregated delays (uSec): avg: 4040, min: 4035, max: 4054, variance: 5
    Rolling average (uSec): 4040

  Current Probe:
    Started at 14:35:38  02 2019 (1 second ago)
    Packets Sent: 1, received: 1
    Measured delays (uSec): avg: 4035, min: 4035, max: 4035, variance: 0
    Probe samples:
            Packet Rx Timestamp Measured Delay
              14:35:38  02 2019 4035081
    Next probe scheduled at 14:36:08  02 2019 (in 29 seconds)
    Next burst packet will be sent in 2 seconds
```

# Additional Commands

## show Commands

***Table 30: show Commands for the Local-End Router (Querier)***

| Command | Description |
|---|---|
| **show performance-measurement summary** [**detail**] | Displays the PM link-delay information, including configuration, session data, and counters. |
| **show performance-measurement interfaces** [**name** *interface-name*] [**detail**] | Displays the PM link-delay information for an interface. |
| **show performance-measurement history interfaces** [**name** *interface-name*] [**adv** \| **aggr** \| **probe**] | • **probe** – Displays the PM link-delay probe history for an interfaces.<br>• **adv** – Displays the PM link-delay advertisement history for interfaces. Advertised values of link delay metrics are values flooded using ISIS, OSPF, or BGP.<br>• **aggr** – Displays the PM link-delay aggregated history for interfaces. |
| **show performance-measurement counters interfaces** [**name** *interface-name*] [**detail**] | Displays the PM link-delay session counters. |
| **show performance-measurement sessions interface** [*session-id*] [**detail**] | Displays information about interfaces that received probe queries from the remote side. |

*Table 31: show Commands for the Remote-End Router (Responder)*

| Command | Description |
|---|---|
| **show performance-measurement responder summary** | Displays the PM for link-delay summary on the remote-end router (responder). |
| **show performance-measurement responder interfaces** [**name** *interface-name*] | Displays the PM link-delay configuration information for interfaces on the remote-end router. |
| **show performance-measurement responder counters interface** [**name** *interface-name*] | Displays the PM link-delay session counters on the remote-end router. |

### clear Commands

*Table 32: clear Commands for the Local-End Router (Querier)*

| Command | Description |
|---|---|
| **clear performance-measurement all** | Clear all performance measurement data, including advertised delay metrics. Using this command withdraws any delay metrics flooded using an IGP or BGP. |
| **clear performance-measurement delay interfaces** [**name** *interface-name*] | Clear PM delay information for interfaces.<br>**Note** Using this command withdraws the previously advertised delay for the cleared interfaces. Use this command with care. |
| **clear performance-measurement counters interfaces** [**name** *interface-name*] | Clear PM interface counters. |
| **clear performance-measurement counters summary** | Clear PM summary counters. |

*Table 33: clear Commands for the Remote-End Router (Responder)*

| Command | Description |
|---|---|
| **clear performance-measurement responder counters interfaces** [**name** *interface-name*] | Clear PM interface counters on the responder. |
| **clear performance-measurement responder counters summary** | Clear PM summary counters on the responder. |

### debug Commands

*Table 34: debug Commands for the Local-End Router (Querier)*

| Command | Description |
|---|---|
| **debug performance-measurement query** [**errors** \| **entry** \| **packet-errors** \| **packets** \| **queues** \| **timers**] | Enable debug messages on the querier. |

*Table 35: debug Commands for the Remote-End Router (Responder)*

| Command | Description |
|---|---|
| **debug performance-measurement responder** [**errors** \| **entry** \| **packet-errors** \| **packets** \| **queues** \| **timers**] | Enable debug messages on the querier. |

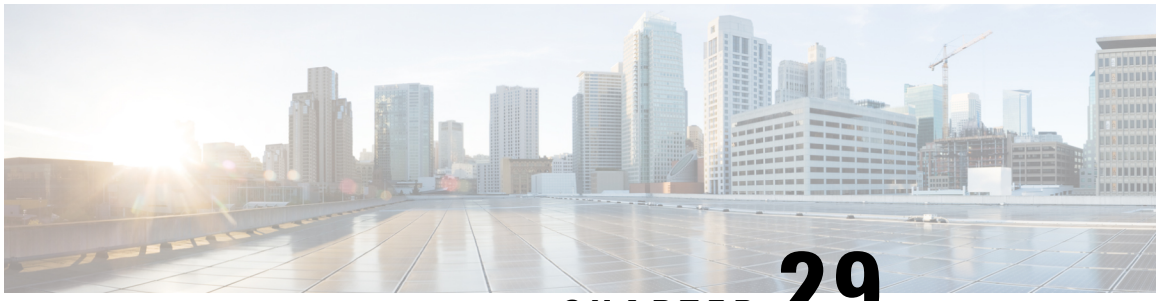
### show tech-support Commands

| Command | Description |
|---|---|
| **show tech-support perf_measure** | Display Performance Measurement related information. |
| **show tech-support monitor event-trace perf_measure** | Display trace information related to Performance Measurement. |

# Additional References

### Standards and RFCs

| Standard/RFC | Title |
|---|---|
| RFC 6374 | Packet Loss and Delay Measurement for MPLS Networks |



CHAPTER 29

# Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains

A border router can advertise the same loopback interface prefixes and the associated prefix Segment Identifiers (SIDs) in multiple ISIS domains.


- Feature Information for Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains, on page 265
- Information about Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains, on page 266
- How to Configure Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains, on page 266
- Example: Configure Loopback Prefix SIDs of a BR in Multiple ISIS Domains, on page 268


## Feature Information for Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains

*Table 36: Feature Information for Performance Measurement for Traffic Engineering*

| Feature Name | Releases | Feature Information |
|---|---|---|
| Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains | Cisco IOS XE Amsterdam 17.3.2 | A border router can advertise loopback interface prefixes and the associated prefix Segment Identifiers (SIDs) in multiple ISIS domains. With such an advertisement, the routers in each associated domain can communicate with the border router using the same prefixes and prefix SIDs. |

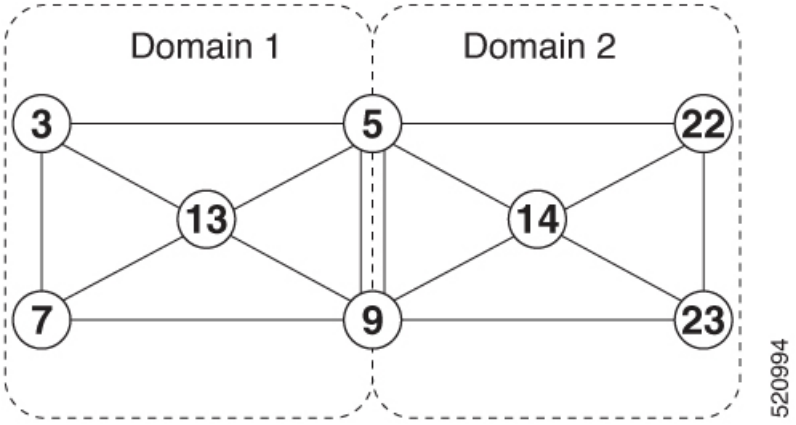
# Information about Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains

## Overview of the Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains

In a segment routing deployment having multiple ISIS domains, it would be beneficial if a border router advertises loopback interface prefixes and prefix SIDs in each associated domain. With such an advertisement, the routers in each associated domain can communicate with the border router using the same prefixes and prefix SIDs.

This feature provides a border router with the capability to advertise prefixes and prefix SIDs into multiple ISIS routing processes, and thereby, into each associated domain.

For example, in the topology shown in the following diagram, the border routers, Router 5 and Router 9, can advertise their prefixes and prefix SIDs in both Domain 1 and Domain 2. A router in Domain 1, say Router 3, and a router in Domain 2, say Router 22, can use the same prefix SIDs to send traffic to send traffic to either border router.




# How to Configure Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains

## Configure the Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains

To advertise a loopback prefix and the prefix SID of a border route in multiple ISIS domains, on the border router, issue the **passive-interface** *loopback-interface-name* command to the ISIS routing process for each domain.

```
router isis 1
  passive-interface loopback 0
router isis 2
  passive-interface loopback 0
```

# Verify the Advertisement of Loopback Prefix SIDs of a Border Router in Multiple ISIS Domains

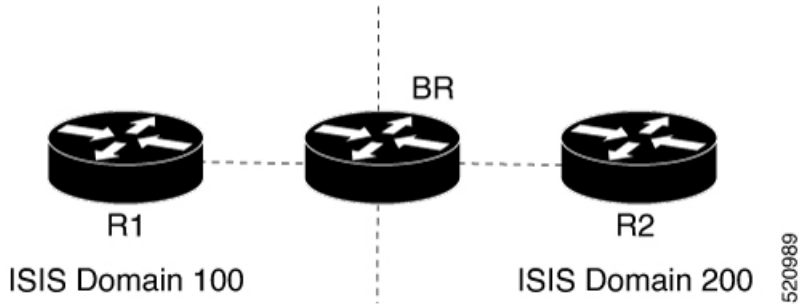
```
Router#show isis database verbose

Tag 1:
IS-IS Level-1 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime/Rcvd      ATT/P/OL
Router.00-00        * 0x00000013   0xDCD8                469/*         0/0/0
  Area Address: 49.0001
  NLPID:        0xCC
  Router CAP:   0.0.0.0, D:0, S:0
    Segment Routing: I:1 V:0, SRGB Base: 16000 Range: 8000
    Segment Routing Local Block: SRLB Base: 15000 Range: 1000
    Segment Routing Algorithms: SPF, Strict-SPF
    Node-MSD
      MSD: 16
  Hostname: Router
  Metric: 0          IP 2.2.2.2/32
    Prefix-attr: X:0 R:0 N:0
  Metric: 0          IP 1.1.1.1/32
    Prefix-attr: X:0 R:0 N:0
    Prefix-SID Index: 1, Algorithm:SPF, R:0 N:1 P:0 E:0 V:0 L:0
IS-IS Level-2 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime/Rcvd      ATT/P/OL
Router.00-00        * 0x00000014   0xDAD9                469/*         0/0/0
  Area Address: 49.0001
  NLPID:        0xCC
  Router CAP:   0.0.0.0, D:0, S:0
    Segment Routing: I:1 V:0, SRGB Base: 16000 Range: 8000
    Segment Routing Local Block: SRLB Base: 15000 Range: 1000
    Segment Routing Algorithms: SPF, Strict-SPF
    Node-MSD
      MSD: 16
  Hostname: Router
  Metric: 0          IP 2.2.2.2/32
    Prefix-attr: X:0 R:0 N:0
  Metric: 0          IP 1.1.1.1/32
    Prefix-attr: X:0 R:0 N:0
    Prefix-SID Index: 1, Algorithm:SPF, R:0 N:1 P:0 E:0 V:0 L:0

Tag 2:
IS-IS Level-1 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime/Rcvd      ATT/P/OL
Router.00-00        * 0x00000012   0xC68A               1179/*         0/0/0
  Area Address: 39.0002
  NLPID:        0xCC
  Router CAP:   1.1.1.1, D:0, S:0
    Segment Routing: I:1 V:0, SRGB Base: 16000 Range: 8000
    Segment Routing Local Block: SRLB Base: 15000 Range: 1000
    Segment Routing Algorithms: SPF, Strict-SPF
    Node-MSD
      MSD: 16
  Hostname: Router
  IP Address:   1.1.1.1
  Metric: 0          IP 1.1.1.1/32
    Prefix-attr: X:0 R:0 N:1
```

```
    Prefix-SID Index: 1, Algorithm:SPF, R:0 N:1 P:0 E:0 V:0 L:0
IS-IS Level-2 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime/Rcvd      ATT/P/OL
Router.00-00        * 0x00000011   0xC889               1184/*         0/0/0
  Area Address: 39.0002
  NLPID:        0xCC
  Router CAP:   1.1.1.1, D:0, S:0
    Segment Routing: I:1 V:0, SRGB Base: 16000 Range: 8000
    Segment Routing Local Block: SRLB Base: 15000 Range: 1000
    Segment Routing Algorithms: SPF, Strict-SPF
    Node-MSD
      MSD: 16
  Hostname: Router
  IP Address:   1.1.1.1
  Metric: 0          IP 1.1.1.1/32
    Prefix-attr: X:0 R:0 N:1
    Prefix-SID Index: 1, Algorithm:SPF, R:0 N:1 P:0 E:0 V:0 L:0
```

# Example: Configure Loopback Prefix SIDs of a BR in Multiple ISIS Domains

The following example shows how to configure a BR and the association of a prefix SID in multiple domains.

Consider the following topology in which we have routers R1 and R2 in two different ISIS domains, and a border router BR that belongs to both the domains.




| Device | Loopback Address | Prefix SID |
|---|---|---|
| R1 | 1.1.1.1/32 | 101 |
| R2 | 2.2.2.2/32 | 202 |
| BR | 3.3.3.3/32 | 303 |

The following configuration on the border router BR causes the router to advertise its loopback interface address and the associated prefix SID in both the connected ISIS domains. This configuration example shows the definition of a loopback interface, the association of a prefix SID with the loopback interface, and the advertisement of the loopback interface address and the associated prefix SID in the ISIS domains ISIS 100 and ISIS 200.

```
BR>enable
BR#configure terminal
BR(config)#interface loopback 0
BR(config-if)#ip address 3.3.3.3 255.255.255.255
BR(config-if)#exit
```

```
BR(config)#segment-routing mpls
BR(config-srmpls)#connected-prefix-sid-map
BR(config-srmpls-conn)#address-family ipv4
BR(config-srmpls-conn-af)#3.3.3.3/32 index 303 range 1
BR(config-srmpls-conn-af)#exit-address-family
BR(config-srmpls-conn-af)#end
BR#configure terminal
BR(config)#router isis 100
BR(config-router)#passive-interface loopback 0
BR(config-router)#exit
BR(config)#router isis 200
BR(config-router)#passive-interface loopback 0
BR(config-router)#end
```