

# Junos® OS

---

## IS-IS User Guide

Published  
2025-12-15

Juniper Networks, Inc.  
1133 Innovation Way  
Sunnyvale, California 94089  
USA  
408-745-2000  
[www.juniper.net](http://www.juniper.net)

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

*Junos® OS IS-IS User Guide*

Copyright © 2025 Juniper Networks, Inc. All rights reserved.

The information in this document is current as of the date on the title page.

## YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

## END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <https://support.juniper.net/support/eula/>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

# Table of Contents

[About This Guide | xiv](#)

1

## [Overview](#)

[Introduction to IS-IS | 2](#)

[IS-IS Overview | 2](#)

[Supported Standards for IS-IS | 7](#)

[IS-IS Fast Reroute Convergence | 10](#)

2

## [Configuring IS-IS](#)

[Configuring a Basic IS-IS Network | 13](#)

[Understanding IS-IS Configuration | 13](#)

[Example: Configuring IS-IS | 15](#)

[Requirements | 15](#)

[Overview | 15](#)

[Configuration | 16](#)

[Verification | 19](#)

[Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups | 21](#)

[Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding | 22](#)

[Requirements | 22](#)

[Overview | 23](#)

[Configuration | 24](#)

[Verification | 30](#)

[Understanding IS-IS Designated Routers | 35](#)

[Configuring Designated Router Election Priority for IS-IS | 36](#)

[Configuring an ISO System Identifier for the Router | 37](#)

[Understanding Default Routes | 37](#)

[How to Configure Multiple Independent IGP Instances of IS-IS and OSPFv2 | 38](#)

[Configure Multiple IGP Instances of IS-IS | 39](#)

**Example: Configure Independent IS-IS Instances in Metro Flooding Domains | 41**

- Overview | 41
- Requirements | 42
- Configuration | 43
- Verification | 60

**Example: Configure Multiple Independent Instances of OSPFv2 with Segment Routing | 67**

- Example Prerequisites | 67
- Before You Begin | 68
- Functional Overview | 68
- Topology Overview | 68
- Topology Illustration | 69
- R2 Configuration Steps | 69
- Verification | 72
- Appendix 1: Set Commands on All Devices | 78

**Configuring IS-IS Authentication and Checksums | 82****Configuring IS-IS Authentication | 82****Configuring IS-IS Authentication Without Network-Wide Deployment | 84****Understanding Hitless Authentication Key Rollover for IS-IS | 84****Example: Configuring Hitless Authentication Key Rollover for IS-IS | 86**

- Requirements | 86
- Overview | 86
- Configuration | 88
- Verification | 93

**Understanding Checksums on IS-IS Interfaces for Error Checking | 94****Example: Enabling Packet Checksums on IS-IS Interfaces for Error Checking | 94**

- Requirements | 95
- Overview | 95
- Configuration | 96
- Verification | 97

**Configuring IS-IS Routing Policy and Route Redistribution | 99****Understanding Routing Policies | 99****Understanding Backup Selection Policy for IS-IS Protocol | 103**

Example: Configuring Backup Selection Policy for IS-IS Protocol | **105**

- Requirements | **105**
- Overview | **105**
- Configuration | **107**
- Verification | **133**

Configuring Backup Selection Policy for the IS-IS Protocol | **144**

Example: Redistributing OSPF Routes into IS-IS | **151**

- Requirements | **152**
- Overview | **152**
- Configuration | **153**
- Verification | **161**

Example: Configuring IS-IS Route Leaking from a Level 2 Area to a Level 1 Area | **165**

- Requirements | **165**
- Overview | **165**
- Configuration | **166**
- Verification | **172**

Handling of the IS-IS Binding SID S Flag and RFC 7794 Prefix Attribute Flags | **174**

Understanding BGP Communities, Extended Communities, and Large Communities as Routing Policy Match Conditions | **176**

Example: Configuring a Routing Policy to Redistribute BGP Routes with a Specific Community Tag into IS-IS | **178**

- Requirements | **178**
- Overview | **178**
- Configuration | **179**
- Verification | **190**

IS-IS Extensions to Support Route Tagging | **191**

Example: Configuring a Routing Policy to Prioritize IS-IS Routes | **193**

- Requirements | **193**
- Overview | **193**
- Configuration | **195**
- Verification | **202**

Configuring Overloading of Stub Networks | **205**

## **Configuring IS-IS Bidirectional Forwarding Detection | 206**

Understanding BFD for IS-IS | 206

Example: Configuring BFD for IS-IS | 209

Requirements | 210

Overview | 210

Configuration | 211

Verification | 215

Understanding BFD Authentication for IS-IS | 217

Configuring BFD Authentication for IS-IS | 219

Configuring BFD Authentication Parameters | 220

Viewing Authentication Information for BFD Sessions | 221

Example: Configuring BFD Authentication for IS-IS | 224

Requirements | 224

Overview | 224

Configuration | 225

Verification | 228

## **Configuring IS-IS Flood Groups | 230**

Understanding IS-IS Flood Group | 230

Example: Configuring IS-IS Flood Group | 231

Requirements | 231

Overview | 231

Configuration | 232

Verification | 235

How to Configure Flood-Reflector Interfaces in IS-IS Networks | 237

Understanding IS-IS Flood Reflectors | 237

Example: IS-IS Flood Reflector | 245

Requirements | 245

Overview | 245

Configuration | 247

Verification | 280

## **Configuring IS-IS Multitopology Routing and IPv6 Support | 287**

IS-IS Multicast Topologies Overview | 287

Example: Configuring IS-IS Multicast Topology | 289

Requirements | 295

Overview | 296

Verification | 297

Understanding Dual Stacking of IPv4 and IPv6 Unicast Addresses | 311

Example: Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses | 313

Requirements | 313

Overview | 313

Configuration | 314

Verification | 318

Understanding IS-IS IPv4 and IPv6 Unicast Topologies | 322

Example: Configuring IS-IS IPv4 and IPv6 Unicast Topologies | 323

Requirements | 323

Overview | 323

Configuration | 325

Verification | 331

**Configuring IS-IS Link and Node Link Protection | 336**

Understanding Loop-Free Alternate Routes for IS-IS | 336

Example: Configuring Node-Link Protection for IS-IS Routes in a Layer 3 VPN | 341

Requirements | 342

Overview | 342

Configuration | 343

Verification | 352

Understanding Remote LFA over LDP Tunnels in IS-IS Networks | 357

Configuring Remote LFA Backup over LDP Tunnels in an IS-IS Network | 359

Example: Configuring Remote LFA over LDP Tunnels in IS-IS Networks | 361

Requirements | 361

Overview | 361

Configuration | 364

Verification | 373

Understanding Weighted ECMP Traffic Distribution on One-Hop IS-IS Neighbors | 378

Example: Weighted ECMP Traffic Distribution on One-Hop IS-IS Neighbors | 379

Requirements | 379

Overview | 380

Configuration | 381

Verification | 396

**Configuring IS-IS Traffic Engineering | 409**

IS-IS Extensions to Support Traffic Engineering | 410

Using Labeled-Switched Paths to Augment SPF to Compute IGP Shortcuts | 411

Example: Enabling IS-IS Traffic Engineering Support | 412

Requirements | 413

Overview | 413

Configuration | 415

Verification | 425

Understanding Forwarding Adjacencies | 433

Example: Advertising Label-Switched Paths into IS-IS | 433

Requirements | 433

Overview | 434

Configuration | 434

Verification | 442

Understanding Wide IS-IS Metrics for Traffic Engineering | 445

Example: Enabling Wide IS-IS Metrics for Traffic Engineering | 446

Requirements | 446

Overview | 446

Configuration | 446

Verification | 448

Understanding LDP-IGP Synchronization | 449

Example: Configuring Synchronization Between IS-IS and LDP | 452

Requirements | 452

Overview | 452

Configuration | 454



Verification | 456

Layer 2 Mapping for IS-IS | 458

Example: Configuring Layer 2 Mapping for IS-IS | 459

Requirements | 460

Overview | 460

Configuration | 461

Verification | 467

Overview of Segment Routing Statistics | 469

Overview of Adjacency Segments, Anycast Segments, and Configurable SRGB in SPRING | 471

Example: Configuring SRGB in Segment Routing for IS-IS | 475

Requirements | 475

Overview | 476

Configuration | 476

Verification | 481

Example: Configuring Anycast and Prefix Segments in SPRING for IS-IS | 483

Requirements | 483

Overview | 484

Configuration | 485

Verification | 498

Configure Segment Routing Global Blocks Label Ranges in SPRING for IS-IS Protocol | 504

Configure Anycast and Prefix segments in SPRING for IS-IS Protocol | 506

Flexible Algorithms in IGP for Segment Routing | 510

Understanding IGP Flexible Algorithms for Segment Routing | 510

Configuring Flexible Algorithm for Segment Routing | 522

Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS and OSPF | 524

Configuring Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS | 529

Example: Configuring Topology Independent Loop-Free Alternate with Segment Routing for IS-IS | 531

Requirements | 531

Overview | 532

Configuration | 533

Verification | 546

Static Adjacency Segment Identifier for IS-IS and OSPF | 549

Overview of Segment Routing over RSVP Forwarding Adjacency in IS-IS | 559

IGP Microloop Avoidance | 560

Overview of SRv6 Network Programming in IS-IS Networks | 563

Example: Configuring SRv6 Network Programming in IS-IS Networks | 570

Requirements | 570

Overview | 570

Configuration | 572

Verification | 589

Understand SRv6 Static Segment Identifier | 601

Binding SID (BSID) for Static SRv6 TE Tunnels | 605

How to Enable Link Delay Measurement and Advertising in IGP | 607

Understanding Link Delay Measurement and Advertising in IGP | 607

Example: Enable IS-IS Link Delay with Source Packet Routing in Networking (SPRING) in a Layer 3 Virtual Private Network (VPN) | 609

Requirements | 610

Overview | 612

Configuration | 613

Verification | 643

Configuring OSPF Link Delay and Delay Normalization on an OSPF Interface | 658

How to Enable Strict SPF SIDs and IGP Shortcut | 663

Understanding Strict SPF (SR-Algo 1) and IGP Shortcuts | 663

Example: Configure Strict SPF SIDs and Enable IGP Shortcuts in SPRING for IS-IS Protocol | 666

Overview | 666

Requirements | 666

Configuration | 668

Verification | 684

Configuring IS-IS Scaling and Throttling | 689

Understanding Link-State PDU Throttling for IS-IS Interfaces | 689

Example: Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces | 690

- Requirements | 690

- Overview | 690

- Configuration | 691

- Verification | 694

Understanding the Transmission Frequency for CSNPs on IS-IS Interfaces | 697

Example: Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces | 698

- Requirements | 698

- Overview | 698

- Configuration | 699

- Verification | 702

Understanding IS-IS Mesh Groups | 705

Example: Configuring Mesh Groups of IS-IS Interfaces | 706

- Requirements | 706

- Overview | 706

- Configuration | 707

- Verification | 711

## Configuring IS-IS CLNS | 715

Understanding IS-IS for CLNS | 715

Example: Configuring IS-IS for CLNS | 715

- Requirements | 716

- Overview | 716

- Configuration | 716

- Verification | 718

## Configuring IS-IS on Logical Systems | 720

Introduction to Logical Systems | 720

Example: Configuring IS-IS on Logical Systems Within the Same Router | 721

- Requirements | 722

- Overview | 722

- Configuration | 723

- Verification | 731

Example: Configuring an IS-IS Default Route Policy on Logical Systems | 736

Requirements | 737

Overview | 737

Configuration | 738

Verification | 742

## Monitoring and Troubleshooting Network Issues

### Monitoring Networks | 747

Example: Tracing Global Routing Protocol Operations | 747

Requirements | 747

Overview | 748

Configuration | 748

Verification | 753

Understanding IS-IS Subscribe Configuration | 754

IS-IS Purge Originator Identification Overview | 755

### Troubleshooting Network Issues | 756

Working with Problems on Your Network | 756

Isolating a Broken Network Connection | 757

Identifying the Symptoms of a Broken Network Connection | 759

Isolating the Causes of a Network Problem | 761

Taking Appropriate Action for Resolving the Network Problem | 762

Evaluating the Solution to Check Whether the Network Problem Is Resolved | 764

### Troubleshooting IS-IS | 766

Verifying the IS-IS Protocol | 766

Verify the LSP | 767

Verify IS-IS Adjacencies and Interfaces | 770

Verify the IS-IS Configuration | 772

Take Appropriate Action | 774

Verify the LSP Again | 776

Verifying the IS-IS Configuration on a Router in a Network | 780

Check the Configuration of a Level 1/Level 2 Router | 782

Check the Configuration of a Level 1 Router | 785

| Check the Configuration of a Level 2 Router | 788

Displaying the Status of IS-IS Adjacencies | 790

| Verifying Adjacent Routers | 792

| Examine the Forwarding Table | 794

Displaying Detailed IS-IS Protocol Information | 795

Analyzing IS-IS Link-State PDUs in Detail | 799

Displaying Sent or Received IS-IS Protocol Packets | 802

4

## **Configuration Statements and Operational Commands**

Junos CLI Reference Overview | 806

# About This Guide

Use this guide to configure, monitor, and troubleshoot the IS-IS routing protocol on your Juniper Network devices.

# 1

PART

## Overview

---

- [Introduction to IS-IS | 2](#)
-

## CHAPTER 1

# Introduction to IS-IS

**IN THIS CHAPTER**

- [IS-IS Overview | 2](#)
- [Supported Standards for IS-IS | 7](#)
- [IS-IS Fast Reroute Convergence | 10](#)

## IS-IS Overview

**IN THIS SECTION**

- [IS-IS Terminology | 3](#)
- [ISO Network Addresses | 3](#)
- [IS-IS Packets | 5](#)
- [Persistent Route Reachability | 6](#)
- [Installing a Default Route to the Nearest Routing Device That Operates at Both IS-IS Levels | 6](#)

The IS-IS protocol is an interior gateway protocol (IGP) that uses link-state information to make routing decisions.

IS-IS is a link-state IGP that uses the shortest-path-first (SPF) algorithm to determine routes. IS-IS evaluates the topology changes and determines whether to perform a full SPF recalculation or a partial route calculation (PRC). This protocol originally was developed for routing International Organization for Standardization (ISO) Connectionless Network Protocol (CLNP) packets.

Like OSPF routing, IS-IS uses hello packets that allow network convergence to occur quickly when network changes are detected. IS-IS uses the SPF algorithm to determine routes. Using SPF, IS-IS evaluates network topology changes and determines if a full or partial route calculation is required.





**NOTE:** Because IS-IS uses ISO addresses, the configuration of IP version 6 (IPv6) and IP version 4 (IPv4) implementations of IS-IS is identical.



**NOTE:** See [Platforms/FPCs That Cannot Forward TCC Encapsulated ISO Traffic](#) to find a list of those devices and FPC configurations that cannot pass ISO traffic when encapsulated in TCC format.

This section discusses the following topics:

## IS-IS Terminology

An IS-IS network is a single autonomous system (AS), also called a *routing domain*, that consists of *end systems* and *intermediate systems*. End systems are network entities that send and receive packets. Intermediate systems send and receive packets and relay (forward) packets. (Intermediate system is the Open System Interconnection [OSI] term for a router.) ISO packets are called network PDUs.

In IS-IS, a single AS can be divided into smaller groups called *areas*. Routing between areas is organized hierarchically, allowing a domain to be administratively divided into smaller areas. This organization is accomplished by configuring *Level 1* and *Level 2* intermediate systems. Level 1 systems route within an area; when the destination is outside an area, they route toward a Level 2 system. Level 2 intermediate systems route between areas and toward other ASs. No IS-IS area functions strictly as a backbone.

Level 1 routers share intra-area routing information, and Level 2 routers share interarea information about IP addresses available within each area. Uniquely, IS-IS routers can act as both Level 1 and Level 2 routers, sharing intra-area routes with other Level 1 routers and interarea routes with other Level 2 routers.

The propagation of link-state updates is determined by the level boundaries. All routers within a level maintain a complete link-state database of all other routers in the same level. Each router then uses the Dijkstra algorithm to determine the shortest path from the local router to other routers in the link-state database.

## ISO Network Addresses

IS-IS uses ISO network addresses. Each address identifies a point of connection to the network, such as a router interface, and is called a *network service access point (NSAP)*.

IS-IS supports multiple NSAP addresses on the loopback lo0 interface.

An end system can have multiple NSAP addresses, in which case the addresses differ only by the last byte (called the *n-selector*). Each NSAP represents a service that is available at that node. In addition to having multiple services, a single node can belong to multiple areas.

Each network entity also has a special network address called a *network entity title (NET)*. Structurally, an NET is identical to an NSAP address but has an n-selector of 00. Most end systems and intermediate systems have one NET. Intermediate systems that participate in multiple areas can have multiple NETs.

The following ISO addresses illustrate the IS-IS address format:

```
49.0001.00a0.c96b.c490.00
49.0001.2081.9716.9018.00
```

NETs take several forms, depending on your network requirements. NET addresses are hexadecimal and range from 8 octets to 20 octets in length. Generally, the format consists of an authority and format Identifier (AFI), a domain ID, an area ID, a system identifier, and a selector. The simplest format omits the domain ID and is 10 octets long. For example, the NET address 49.0001.1921.6800.1001.00 consists of the following parts:

- 49—AFI
- 0001—Area ID
- 1921.6800.1001—System identifier
- 00—Selector

The system identifier must be unique within the network. For an IP-only network, we recommend using the IP address of an interface on the router. Configuring a loopback NET address with the IP address is helpful when troubleshooting is required on the network.

The first portion of the address is the area number, which is a variable number from 1 through 13 bytes. The first byte of the area number (49) is the authority and format indicator (AFI). The next bytes are the assigned domain (area) identifier, which can be from 0 through 12 bytes. In the examples above, the area identifier is 0001.

The next six bytes form the system identifier. The system identifier can be any six bytes that are unique throughout the entire domain. The system identifier commonly is the media access control (MAC) address (as in the first example, 00a0.c96b.c490) or the IP address expressed in binary-coded decimal (BCD) (as in the second example, 2081.9716.9018, which corresponds to IP address 208.197.169.18). The last byte (00) is the n-selector.



**NOTE:** The system identifier cannot be 0000.0000.0000. All 0s is an illegal setting, and the adjacency is not formed with this setting.

To provide help with IS-IS debugging, the Junos® operating system (Junos OS) supports dynamic mapping of ISO system identifiers to the hostname. Each system can be configured with a hostname, which allows the system identifier-to-hostname mapping to be carried in a dynamic hostname type,

length, and value (TLV) tuple in IS-IS link-state PDUs. This enables intermediate systems in the routing domain to learn about the ISO system identifier of a particular intermediate system.

## IS-IS Packets

Each IS-IS PDU shares a common header. IS-IS uses the following PDUs to exchange protocol information:

- IS-IS hello (IIH) PDUs—Broadcast to discover the identity of neighboring IS-IS systems and to determine whether the neighbors are Level 1 or Level 2 intermediate systems.

IS-IS hello PDUs establish adjacencies with other routers and have three different formats: one for point-to-point hello packets, one for Level 1 broadcast links, and one for Level 2 broadcast links. Level 1 routers must share the same area address to form an adjacency, while Level 2 routers do not have this limitation. The request for adjacency is encoded in the Circuit type field of the PDU.

Hello PDUs have a preset length assigned to them. The IS-IS router does not resize any PDU to match the maximum transmission unit (MTU) on a router interface. Each interface supports the maximum IS-IS PDU of 1492 bytes, and hello PDUs are padded to meet the maximum value. When the hello is sent to a neighboring router, the connecting interface supports the maximum PDU size.

- Link-state PDUs—Contain information about the state of adjacencies to neighboring IS-IS systems. Link-state PDUs are flooded periodically throughout an area.

Also included is metric and IS-IS neighbor information. Each link-state PDU must be refreshed periodically on the network and is acknowledged by information within a sequence number PDU.

On point-to-point links, each link-state PDU is acknowledged by a partial sequence number PDU (PSNP), but on broadcast links, a complete sequence number PDU (CSNP) is sent out over the network. Any router that finds newer link-state PDU information in the CSNP then purges the out-of-date entry and updates the link-state database.

Link-state PDUs support variable-length subnet mask addressing.

- Complete sequence number PDUs (CSNPs)—Contain a complete list of all link-state PDUs in the IS-IS database. CSNPs are sent periodically on all links, and the receiving systems use the information in the CSNP to update and synchronize their link-state PDU databases. The designated router multicasts CSNPs on broadcast links in place of sending explicit acknowledgments for each link-state PDU.

Contained within the CSNP is a link-state PDU identifier, a lifetime, a sequence number, and a checksum for each entry in the database. Periodically, a CSNP is sent on both broadcast and point-to-point links to maintain a correct database. Also, the advertisement of CSNPs occurs when an adjacency is formed with another router. Like IS-IS hello PDUs, CSNPs come in two types: Level 1 and Level 2.

When a device receives a CSNP, it checks the database entries against its own local link-state database. If it detects missing information, the device requests specific link-state PDU details using a partial sequence number PDU (PSNP).

- Partial sequence number PDUs (PSNPs)—Sent multicast by a receiver when it detects that it is missing a link-state PDU (when its link-state PDU database is out of date). The receiver sends a PSNP to the system that transmitted the CSNP, effectively requesting that the missing link-state PDU be transmitted. That routing device, in turn, forwards the missing link-state PDU to the requesting routing device.

A PSNP is used by an IS-IS router to request link-state PDU information from a neighboring router. A PSNP can also explicitly acknowledge the receipt of a link-state PDU on a point-to-point link. On a broadcast link, a CSNP is used as implicit knowledge. Like hello PDUs and CSNPs, the PSNP also has two types: Level 1 and Level 2.

When a device compares a CSNP to its local database and determines that a link-state PDU is missing, the router issues a PSNP for the missing link-state PDU, which is returned in a link-state PDU from the router sending the CSNP. The received link-state PDU is then stored in the local database, and an acknowledgment is sent back to the originating router.

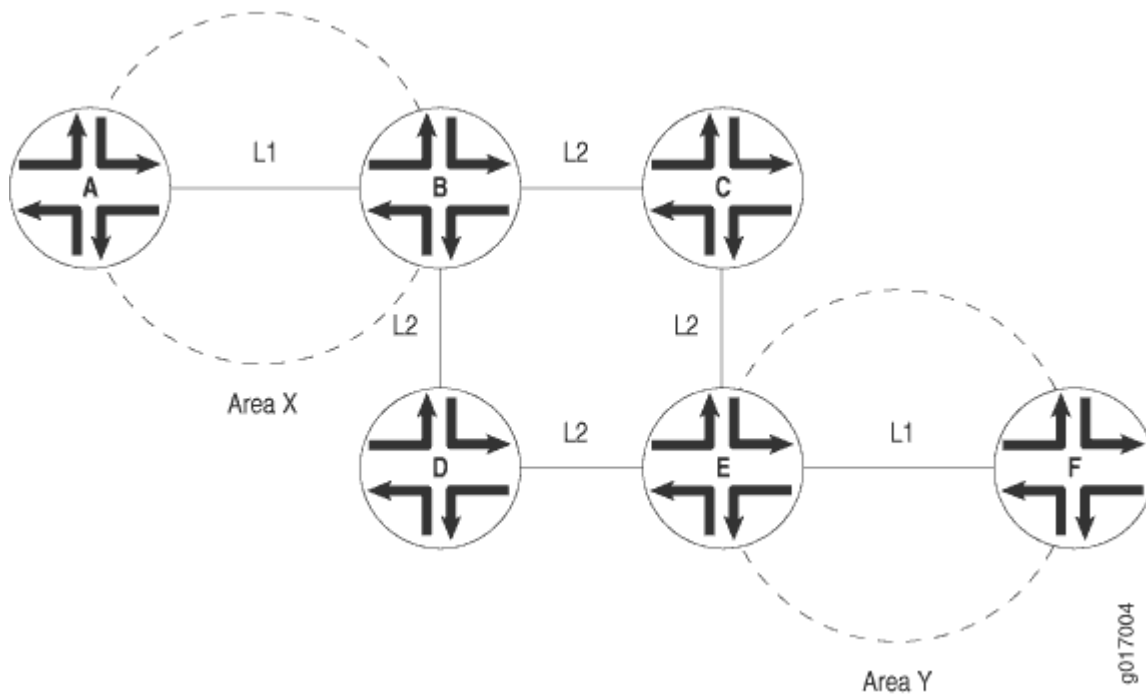
## Persistent Route Reachability

IPv4 and IPv6 route reachability information in IS-IS link-state PDUs is preserved when you commit a configuration. IP prefixes are preserved with their original packet fragment upon link-state PDU regeneration.

## Installing a Default Route to the Nearest Routing Device That Operates at Both IS-IS Levels

When a routing device that operates as both a Level 1 and Level 2 router (Router B) determines that it can reach at least one area other than its own (for example, in Area Y), it sets the ATTACHED bit in its Level 1 link-state PDU. Thereafter, the Level 1 router (Router A) introduces a default route pointing to the nearest attached routing device that operates as both a Level 1 and Level 2 router (Router B). See [Figure 1 on page 7](#).

Figure 1: Install Default Route to Nearest Routing Device That Operates at Both Level 1 and Level 2



## RELATED DOCUMENTATION

[Understanding IS-IS Configuration | 13](#)

[Example: Configuring IS-IS | 15](#)

## Supported Standards for IS-IS

Junos OS substantially supports the following standards for IS-IS.

- International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 8473, *Information technology – Protocol for providing the connectionless-mode network service*
- ISO 9542, *End System to Intermediate System Routing Exchange Protocol for Use in Conjunction with the Protocol for the Provision of the Connectionless-mode Network Service*
- ISO/IEC 10589, *Information technology – Telecommunications and information exchange between systems – Intermediate System to Intermediate System intra-domain routing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode network service (ISO 8473)*

- RFC 1195, *Use of OSI IS-IS for Routing in TCP/IP and Dual Environments*
- RFC 5120, *M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)*
- RFC 5130, *A Policy Control Mechanism in IS-IS Using Administrative Tags*
- RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*
- RFC 5301, *Dynamic Hostname Exchange Mechanism for IS-IS*
- RFC 5302, *Domain-Wide Prefix Distribution with Two-Level IS-IS*
- RFC 5303, *Three-Way Handshake for IS-IS Point-to-Point Adjacencies*
- RFC 5304, *IS-IS Cryptographic Authentication*
- RFC 5305, *IS-IS Extensions for Traffic Engineering*
- RFC 5306, *Restart Signaling for IS-IS*
- RFC 5307, *IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*
- RFC 5308, *Routing IPv6 with IS-IS*
- RFC 5310, *IS-IS Generic Cryptographic Authentication*
- RFC 5880, *Bidirectional Forwarding Detection (BFD)*
- RFC 6119, *IPv6 Traffic Engineering in IS-IS*
- RFC 6232, *Purge Originator Identification TLV for IS-IS*
- RFC 6233, *IS-IS Registry Extension for Purges*
- RFC 7775, *IS-IS Route Preference for Extended IP and IPv6 Reachability*
- RFC 7794, *IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability*
- RFC 7981, *IS-IS Extensions for Advertising Router Information*
- RFC 8202, *IS-IS Multi-Instance*
- RFC 8518, *Selection of Loop-Free Alternates for Multi-Homed Prefixes*
- RFC 8570, *IS-IS Traffic Engineering (TE) Metric Extensions*
- RFC 8667, *IS-IS Extensions for Segment Routing*
- RFC 8706, *Restart Signaling for IS-IS*

- RFC 8919, *IS-IS Application-Specific Link Attributes*
- RFC 9350 *IGP Flexible Algorithm* (Partial support)
- RFC 9352, *IS-IS Extensions to Support Segment Routing over the IPv6 Data Plane*

The following RFCs do not define standards, but provide information about IS-IS and related technologies. The IETF classifies them as “Informational.”

- RFC 2104, *HMAC: Keyed-Hashing for Message Authentication*
- RFC 2973, *IS-IS Mesh Groups*
- RFC 3277, *Intermediate System to Intermediate System (IS-IS) Transient Blackhole Avoidance*
- RFC 3358, *Optional Checksums in Intermediate System to Intermediate System (ISIS)*
- RFC 3359, *Reserved Type, Length and Value (TLV) Codepoints in Intermediate System to Intermediate System*
- RFC 3373, *Three-Way Handshake for Intermediate System to Intermediate System (IS-IS) Point-to-Point Adjacencies*
- RFC 3567, *Intermediate System to Intermediate System (IS-IS) Cryptographic Authentication*
- RFC 3719, *Recommendations for Interoperable Networks using Intermediate System to Intermediate System (IS-IS)*
- RFC 3787, *Recommendations for Interoperable IP Networks using Intermediate System to Intermediate System (IS-IS)*
- RFC 3847, *Restart Signaling for Intermediate System to Intermediate System (IS-IS)*
- RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*
- RFC 6151, *updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms*
- Internet draft draft-ietf-isis-wg-255adj-02.txt, *Maintaining more than 255 circuits in IS-IS*
- Internet draft draft-przygienda-flood-reflector-00, *Flood Reflectors*
- Internet draft draft-przygienda-lsr-flood-reflection-01, *IS-IS Flood Reflection*

## RELATED DOCUMENTATION

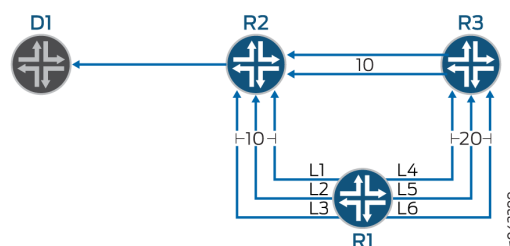
## IS-IS Fast Reroute Convergence

Subsecond service restoration is a key requirement for MPLS and native IP-based network service providers. There are many ways to achieve fast reroute with a sub-optimal next-hop to reach the destination, such as loop-free alternate and remote loop-free alternate. In these cases, IGP downloads the primary and backup next-hop beforehand in the forwarding information base (FIB). A packet forwarding engine (PFE) performs a local repair when the primary next-hop loses its reachability to a given destination. Since the PFE already has an alternative path to reach its destination, subsecond restoration is possible. If the destination is reachable through equal-cost multi-path (ECMP), only the primary path is downloaded to the FIB. If a few ECMP links go down below the required bandwidth for a destination, fast reroute convergence is not possible.

To resolve this, the best ECMP links are grouped as a unilist of primary next-hops to reach the destination, and the sub-optimal ECMP links are grouped as a unilist of backup next-hops to reach the destination. If the bandwidth of the primary next-hops falls below the desired bandwidth, the PFE does a local repair and switches traffic to backup unilist next-hops. This is yet another backup, where the backup path is computed and installed in FIB for ECMP paths. Here, a set of best ECMP links is grouped as primary next-hops to reach the destination, and a set of sub-optimal ECMP links is grouped as backup next-hops to reach the destination. If the bandwidth of the primary next-hops falls below the desired bandwidth due to link failure on the primary group, the PFE should perform a local repair and switch the traffic to backup next-hops.

In the following topology, R1 has three ECMP links to D1 via R2. R1 also has three sub-optimal ECMP links to D1 via R3 and R2. All ECMP links L1, L2, and L3 can be placed under one group; a primary group, and also group sub-optimal ECMP links L3, L4, and L5 under another group, a backup group.

**Figure 2: Topology**





IS-IS calculates the shortest path using the shortest-path-first (SPF) algorithm and downloads primary next-hops with appropriate weight in FIB. IS-IS also calculates backup next-hops and downloads them to FIB with appropriate weight.

Backup next-hop weight will always be greater than the primary next-hop weight. If a link from the primary group goes down, the PFE performs a local repair and modifies the weight of the next-hops. The PFE forwards traffic to the destination with the least weight next-hops to achieve sub-millisecond convergence. IS-IS runs SPF and comes up with a set of primary and backup next-hops. IS-IS then updates the FIB with the updated next hops. PFE resumes traffic forwarding on new next-hops without any traffic loss.

## RELATED DOCUMENTATION

| *link-group-protection (Protocols IS-IS)*

# 2

PART

## Configuring IS-IS

---

- [Configuring a Basic IS-IS Network | 13](#)
  - [Configuring IS-IS Authentication and Checksums | 82](#)
  - [Configuring IS-IS Routing Policy and Route Redistribution | 99](#)
  - [Configuring IS-IS Bidirectional Forwarding Detection | 206](#)
  - [Configuring IS-IS Flood Groups | 230](#)
  - [Configuring IS-IS Multitopology Routing and IPv6 Support | 287](#)
  - [Configuring IS-IS Link and Node Link Protection | 336](#)
  - [Configuring IS-IS Traffic Engineering | 409](#)
  - [Configuring IS-IS Scaling and Throttling | 689](#)
  - [Configuring IS-IS CLNS | 715](#)
  - [Configuring IS-IS on Logical Systems | 720](#)
-

## CHAPTER 2

# Configuring a Basic IS-IS Network

## IN THIS CHAPTER

- Understanding IS-IS Configuration | 13
- Example: Configuring IS-IS | 15
- Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups | 21
- Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding | 22
- Understanding IS-IS Designated Routers | 35
- Configuring Designated Router Election Priority for IS-IS | 36
- Configuring an ISO System Identifier for the Router | 37
- Understanding Default Routes | 37
- How to Configure Multiple Independent IGP Instances of IS-IS and OSPFv2 | 38

## Understanding IS-IS Configuration

To configure IS-IS, you must enable IS-IS on the interfaces and configure a NET address on one of the device interfaces (preferably, the lo0 interface) by setting `family iso address net-address` on the interface. To create the NET address (also known as the system ID or the NSAP address), you can use the convention that is dictated by your network design, or you can follow this convention:

1. Take the router ID, remove the dots (.), and insert leading zeroes where necessary so that the string is 12 characters long.

For example, if the router ID is 192.168.0.4, the 12-character string would be 192168000004. If the router ID is 10.12.23.1, the 12-character string would be 010012023001.

2. Add a dot after every 4th character.

The strings would become 1921.6800.0004 and 0100.1202.3001.

3. Prepend the area number.

If the routing devices are in area 47, the strings would become 47.1921.6800.0004 and 47.0100.1202.3001.

#### 4. Append the selector (00).

The strings would become 47.1921.6800.0004.00 and 47.0100.1202.3001.00.

You must configure the ISO family on all interfaces that are supporting the IS-IS protocol by setting `family iso` on the interface. This means that IS-IS related frames are not discarded by the routing devices.

You must enable IS-IS to run on the interfaces by setting `interface interface-name` in the protocol configuration. This means that the interfaces are advertised into IS-IS.

Unlike OSPF, when you enable IS-IS on the lo0 interface, you do not need to explicitly set passive mode. Passive mode means that the interface is advertised into the link-state protocol, but the interface does not send or receive protocol control packets, such as IS-IS hello and link-state PDUs. In IS-IS, the lo0 interface is always passive.

When you enable IS-IS on an interface, both levels (Level 1 and Level 2) are enabled by default. To specify that an interface is on a Level 1 link, disable Level 2. To specify that an interface is on a Level 2 link, disable Level 1. You can disable a level on the entire device or per-interface. If two routing devices, R1 and R2, are both in the same IS-IS area, they communicate at Level 1 if one or both devices have Level 2 disabled.

For security devices only, you must enable IS-IS by setting `mode packet-based` at the `[edit security forwarding-options family iso]` hierarchy level.



**NOTE:** Junos releases prior to and including 19.2R1-S2 supported ISIS PDU exchange without explicitly setting packet mode for ISIS under `[edit security forwarding-options family iso]`. In newer releases this setting is required in order for ISIS to operate properly. When upgrading an SRX device to a Junos version newer than 19.2R1-S2 you must configure packet mode for ISIS or adjacencies will not form.

## RELATED DOCUMENTATION

[Example: Configuring IS-IS | 15](#)

[Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding | 22](#)

[Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups | 21](#)

## Example: Configuring IS-IS

### IN THIS SECTION

- Requirements | 15
- Overview | 15
- Configuration | 16
- Verification | 19

This example shows how to configure IS-IS in a simple two-device network topology.



**NOTE:** Our content testing team has validated and updated this example.

### Requirements

No special configuration beyond device initialization is required before configuring this example.



**NOTE:** Are you interested in getting hands-on experience on this feature?

Visit Juniper vLabs to reserve your pre-configured [vLab Sandbox: IS-IS - Single-area](#) and try it for free!

### Overview

In this example, you configure two devices, Device R1 and Device R2, within a single IS-IS area.

To enable IS-IS routing, you must:

- Configure a NET address (also known as the system ID or the NSAP address) on one of the device interfaces (preferably, the lo0 interface) by including the `family iso address net-address` statement on the interface.
- Configure the ISO family on all interfaces that are supporting the IS-IS protocol by including the `family iso` statement on the interface.
- Advertise the device interfaces into IS-IS by including the `interface interface-name` statement in the protocol configuration.

- (On security devices only) Enable the forwarding of IS-IS traffic by including the family mode packet-based statement in the security forwarding options configuration.



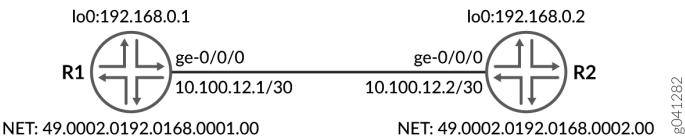
**NOTE:** Junos releases prior to and including 19.2R1-S2 supported ISIS PDU exchange without explicitly setting packet mode for ISIS under [edit security forwarding-options family iso]. In newer releases this setting is required in order for ISIS to operate properly. When upgrading an SRX device to a Junos version newer than 19.2R1-S2 you must configure packet mode for ISIS or adjacencies will not form.

Figure 3 on page 16 shows the topology used in this example.



**NOTE:** In this example, we use NET addresses that incorporate the router's loopback IP addresses (49.0002.0192.0168.0001.00 and 49.0002.0192.0168.0002.00) for readability and learning purposes. In production environments, we recommend that you derive the system identifier portion of the NET address from the router's MAC address or another unique hardware identifier. For more information about configuring proper NET addresses, see ["Configuring an ISO System Identifier for the Router" on page 37](#).

Figure 3: Simple IS-IS Topology



"CLI Quick Configuration" on page 17 shows the configuration for both of the devices in Figure 3 on page 16.

## Configuration

### IN THIS SECTION

- Procedure | 17

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R1

```
set security forwarding-options family iso mode packet-based
set interfaces ge-0/0/0 unit 0 description "To R2"
set interfaces ge-0/0/0 unit 0 family inet address 10.100.12.1/30
set interfaces ge-0/0/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set protocols isis interface ge-0/0/0.0
set protocols isis interface lo0.0
```

#### Device R2

```
set security forwarding-options family iso mode packet-based
set interfaces ge-0/0/0 unit 0 description "To R1"
set interfaces ge-0/0/0 unit 0 family inet address 10.100.12.2/30
set interfaces ge-0/0/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0002.00
set protocols isis interface ge-0/0/0.0
set protocols isis interface lo0.0
```

### Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure IS-IS:

1. (On security devices only) Enable the forwarding of IS-IS traffic to overcome the default behavior of dropping IS-IS traffic.

```
[edit security]
user@R1# set forwarding-options family iso mode packet-based
```

2. Configure the device interface, and enable the ISO family on the interface.

```
[edit interfaces]
user@R1# set ge-0/0/0 unit 0 description "To R2"
user@R1# set ge-0/0/0 unit 0 family inet address 10.100.12.1/30
user@R1# set ge-0/0/0 unit 0 family iso
```

3. Configure the loopback interface and set the NET address.

```
[edit interfaces]
user@R1# set lo0 unit 0 family inet address 192.168.0.1/32
user@R1# set lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
```

4. Enable IS-IS for the device interfaces.

```
[edit protocols]
user@R1# set isis interface ge-0/0/0.0
user@R1# set isis interface lo0.0
```

## Results

From configuration mode, confirm your configuration by entering the `show` command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show
security {
  forwarding-options {
    family iso {
      mode packet-based;
    }
  }
}
```



```

}
interfaces {
  ge-0/0/0 {
    unit 0 {
      description "To R2";
      family inet {
        address 10.100.12.1/30;
      }
      family iso;
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 192.168.0.1/32;
      }
      family iso {
        address 49.0002.0192.0168.0001.00;
      }
    }
  }
}
protocols {
  isis {
    interface ge-0/0/0.0;
    interface lo0.0;
  }
}

```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Verifying IS-IS Adjacencies and Interfaces | 20](#)
- [Verifying End-to-End Connectivity | 20](#)

Confirm that the configuration is working properly.

## Verifying IS-IS Adjacencies and Interfaces

### Purpose

Verify that IS-IS adjacencies are up and that the IS-IS interfaces are included in the protocol configuration.

### Action

From operational mode, enter the `show isis adjacency` and `show isis interface` commands.

```
user@R1> show isis adjacency
```

Interface	System	L State	Hold (secs)	SNPA
ge-0/0/0.0	R2	1 Up	7	56:4:1e:0:5f:58
ge-0/0/0.0	R2	2 Up	7	56:4:1e:0:5f:58

```
user@R1> show isis interface
```

IS-IS interface database:

Interface	L	CirID	Level 1 DR	Level 2 DR	L1/L2 Metric
ge-0/0/0.0	3	0x1	R2.02	R2.02	10/10
lo0.0	3	0x1	Passive	Passive	0/0

### Meaning

Device R1 has established adjacency with Device R2 as indicated by the State output field which is Up.

Device R1 interfaces are advertised into IS-IS with Device R2 as the designated router responsible for sending link-state advertisements. Both the devices are enabled with Level 1 and Level 2 IS-IS as indicated by the L 3 output field.

## Verifying End-to-End Connectivity

### Purpose

Verify that the devices are reachable by pinging their loopback addresses.

## Action

From operational mode, enter the `ping 192.168.0.1 source 192.168.0.2 count 100 rapid` command.

```
user@R1> ping 192.168.0.1 source 192.168.0.2 count 100 rapid
PING 192.168.0.2 (192.168.0.2): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!
--- 192.168.0.2 ping statistics ---
100 packets transmitted, 100 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.807/2.425/6.447/0.553 ms
```

## Meaning

The devices can successfully ping each other's loopback address.

## RELATED DOCUMENTATION

[Understanding IS-IS Configuration | 13](#)

[Configuring Designated Router Election Priority for IS-IS | 36](#)

[Verifying the IS-IS Protocol](#)

[Configuring Graceful Restart Options for IS-IS](#)

[Configuring an ISO System Identifier for the Router | 37](#)

## Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups

In IS-IS, a single AS can be divided into smaller groups called *areas*.

Link-state protocols cannot scale well if a large autonomous system (AS) consists of a single set of routing devices that all share a common database to compute the best paths through the AS. Because the shortest-path-first (SPF) algorithm works in an exponential fashion, the CPU demand can become too heavy when too many routing devices share their complete routing information with each other. To alleviate this issue, large ASs are divided into smaller parts called areas.

When ASs are split into areas, the disjointed areas must be connected to route traffic between the areas. Reachability information at the area borders must be injected into each other areas.

In IS-IS, routing between areas is organized hierarchically. This organization is accomplished by configuring *Level 1* and *Level 2* intermediate systems. Level 1 systems route within an area. When the destination is outside an area, Level 1 systems route toward a Level 2 system. Level 2 intermediate systems route between areas and toward other ASs. No IS-IS area functions strictly as a backbone.

Level 1 routers share intra-area routing information, and Level 2 routers share interarea information about IP addresses available within each area. Uniquely, IS-IS routers can act as both Level 1 and Level 2 routers, sharing intra-area routes with other Level 1 routers and interarea routes with other Level 2 routers.

The propagation of link-state updates is determined by the level boundaries. All routers within a level maintain a complete link-state database of all other routers in the same level. Each router then uses the Dijkstra algorithm to determine the shortest path from the local router to other routers in the link-state database.

## RELATED DOCUMENTATION

| [Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding](#) | 22

## Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding

### IN THIS SECTION

- [Requirements](#) | 22
- [Overview](#) | 23
- [Configuration](#) | 24
- [Verification](#) | 30

This example shows how to configure a multi-level IS-IS topology.

### Requirements

No special configuration beyond device initialization is required before configuring this example.

## Overview

### IN THIS SECTION

- [Topology | 24](#)

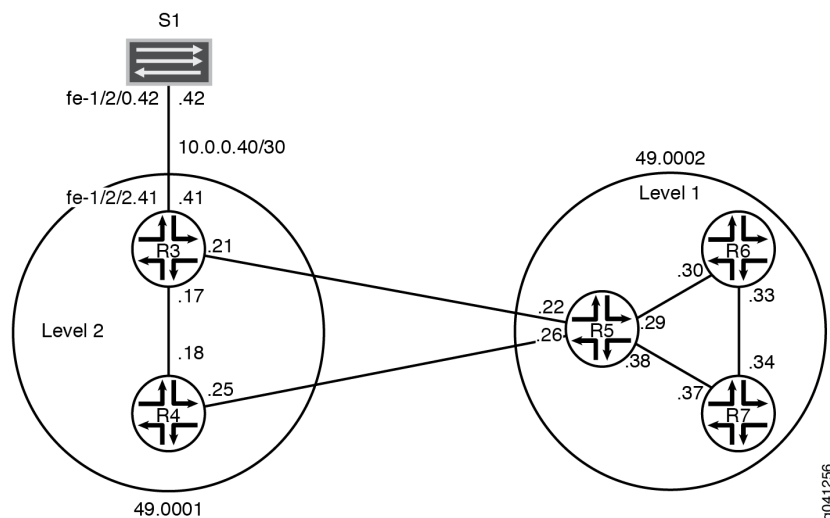
Like OSPF, the IS-IS protocol supports the partitioning of a routing domain into multiple areas with levels that control interarea flooding. The use of multiple levels improves protocol scalability, as Level 2 (backbone) link-state PDUs are normally not flooded into a Level 1 area.

An IS-IS Level 2 area is analogous to the OSPF backbone area (0), while a Level 1 area operates much like an OSPF totally stubby area, in that a default route is normally used to reach both inter-level and AS external routes.

Unlike OSPF, IS-IS area boundaries occur between routers, such that a given routing device is always wholly contained within a particular area. Level 1 adjacencies can be formed between routers that share a common area number, while a Level 2 adjacency can be formed between routers that might or might not share an area number.

[Figure 4 on page 23](#) shows the topology used in this example.

**Figure 4: IS-IS Multi-Level Topology**



"CLI Quick Configuration" on [page 24](#) shows the configuration for all of the devices in [Figure 4 on page 23](#). The section "No Link Title" on [page 27](#) describes the steps on Device R5.

This example has the following characteristics:

- Device R5 functions as a Level 1/Level 2 router to interconnect the Level 2 backbone area 49.0001 and the Level 1 area 49.0002 containing Device R6 and Device R7.
- The system ID is based on the devices' IPv4 lo0 addresses.
- Loss of any individual interface does not totally disrupt the IS-IS operation.
- The IPv4 lo0 addresses of all routers are reachable through IS-IS.
- The link between Device R3 and Device S1 appears in area 49.0001 as an intra-area route. No IS-IS adjacencies can be established on this interface. This is accomplished by configuring the `passive` statement on Device R3's interface to Device S1.
- The loopback addresses of Level 2 devices do not appear in a Level 1 area.
- There is only one adjacency for each device pairing.

## Topology

## Configuration

### IN THIS SECTION

- [Procedure | 24](#)

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the `[edit]` hierarchy level.

#### Device R3

```
set interfaces fe-1/2/0 unit 0 description to-R4
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.17/30
set interfaces fe-1/2/0 unit 0 family iso
```

```

set interfaces fe-1/2/1 unit 0 description to-R5
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.21/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.41/30
set interfaces fe-1/2/2 unit 0 description to-S1
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0001.1921.6800.0003.00
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0 level 1 disable
set protocols isis interface fe-1/2/2.0 passive

```

#### Device R4

```

set interfaces fe-1/2/0 unit 0 description to-R3
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.18/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R5
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.25/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.4/32
set interfaces lo0 unit 0 family iso address 49.0001.0192.0168.0004.00
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0 level 1 disable

```

#### Device R5

```

set interfaces fe-1/2/0 unit 0 description to-R3
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.22/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R4
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.26/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/2 unit 0 description to-R6
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.29/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces fe-1/2/3 unit 0 description to-R7
set interfaces fe-1/2/3 unit 0 family inet address 10.0.0.38/30

```

```

set interfaces fe-1/2/3 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.5/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0005.00
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface fe-1/2/2.0 level 2 disable
set protocols isis interface fe-1/2/3.0 level 2 disable
set protocols isis interface lo0.0 level 1 disable

```

## Device R6

```

set interfaces fe-1/2/0 unit 0 description to-R5
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.30/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R7
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.33/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.6/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0006.00
set protocols isis interface fe-1/2/0.0 level 2 disable
set protocols isis interface fe-1/2/1.0 level 2 disable
set protocols isis interface lo0.0 level 2 disable

```

## Device R7

```

set interfaces fe-1/2/0 unit 0 description to-R6
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.34/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R5
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.37/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.7/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0007.00
set protocols isis interface fe-1/2/0.0 level 2 disable
set protocols isis interface fe-1/2/1.0 level 2 disable
set protocols isis interface lo0.0 level 2 disable

```



## Device S1

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.42/30
set interfaces fe-1/2/0 unit 0 description to-R3
```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure multi-level IS-IS:

### 1. Configure the network interfaces.

Enable IS-IS on the interfaces by Including the ISO address family on each interface.

```
[edit interfaces]
user@R5# set fe-1/2/0 unit 0 description to-R3
user@R5# set fe-1/2/0 unit 0 family inet address 10.0.0.22/30
user@R5# set fe-1/2/0 unit 0 family iso
user@R5# set fe-1/2/1 unit 0 description to-R4
user@R5# set fe-1/2/1 unit 0 family inet address 10.0.0.26/30
user@R5# set fe-1/2/1 unit 0 family iso
user@R5# set fe-1/2/2 unit 0 description to-R6
user@R5# set fe-1/2/2 unit 0 family inet address 10.0.0.29/30
user@R5# set fe-1/2/2 unit 0 family iso
user@R5# set fe-1/2/3 unit 0 description to-R7
user@R5# set fe-1/2/3 unit 0 family inet address 10.0.0.38/30
user@R5# set fe-1/2/3 unit 0 family iso
```

### 2. Configure two loopback interface addresses.

One address is for IPv4.

The other is for the IS-IS area 49.0002 so that Device R5 can form adjacencies with the other Level 1 devices in area 49.0002. Even though Device R5's NET identifies itself as belonging to the Level 1

area 49.0002, its loopback interface is not configured as a Level 1 interface. Doing so would cause the route to Device R5's loopback to be injected into the Level 1 area.

```
[edit interfaces lo0 unit 0]
user@R5# set family inet address 192.168.0.5/32
user@R5# set family iso address 49.0002.0192.0168.0005.00
```

### 3. Specify the IS-IS level on a per-interface basis.

Device R5 becomes adjacent to the other routing devices on the same level on each link.

By default, IS-IS is enabled for IS-IS areas on all interfaces on which the ISO protocol family is enabled (at the `[edit interfaces interface-name unit logical-unit-number]` hierarchy level). To disable IS-IS at any particular level on an interface, include the `disable` statement.

Device R5's loopback interface is configured to run Level 2 only. If Level 1 operation were enabled on lo0.0, Device R5 would include its loopback address in its Level 1 link-state PDU, which is incorrect for this example in which the loopback addresses of Level 2 devices must not appear in a Level 1 area.

Unlike OSPF, you must explicitly list the router's lo0 interface at the `[edit protocols isis]` hierarchy level, because this interface is the source of the router's NET, and therefore must be configured as an IS-IS interface. In IS-IS, the lo0 interface operates in the passive mode by default, which is ideal because adjacency formation can never occur on a virtual interface.

```
[edit protocols isis]
user@R5# set interface fe-1/2/0.0 level 1 disable
user@R5# set interface fe-1/2/1.0 level 1 disable
user@R5# set interface fe-1/2/2.0 level 2 disable
user@R5# set interface fe-1/2/3.0 level 2 disable
user@R5# set interface lo0.0 level 1 disable
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces` and `show protocols` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R5# show interfaces
fe-1/2/0 {
  unit 0{
```

```

        description to-R3;
        family inet {
            address 10.0.0.22/30;
        }
        family iso;
    }
}
fe-1/2/1 {
    unit 0 {
        description to-R4;
        family inet {
            address 10.0.0.26/30;
        }
        family iso;
    }
}
fe-1/2/2 {
    unit 0 {
        description to-R6;
        family inet {
            address 10.0.0.29/30;
        }
        family iso;
    }
}
fe-1/2/3 {
    unit 0 {
        description to-R7;
        family inet {
            address 10.0.0.38/30;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.0.5/32;
        }
        family iso {
            address 49.0002.0192.0168.0005.00;
        }
    }
}

```

```
}  
}
```

```
user@R5# show protocols  
isis {  
  interface fe-1/2/0.0 {  
    level 1 disable;  
  }  
  interface fe-1/2/1.0 {  
    level 1 disable;  
  }  
  interface fe-1/2/2.0 {  
    level 2 disable;  
  }  
  interface fe-1/2/3.0 {  
    level 2 disable;  
  }  
  interface lo0.0 {  
    level 1 disable;  
  }  
}
```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Checking Interface-to-Area Associations | 31](#)
- [Verifying IS-IS Adjacencies | 31](#)
- [Examining the IS-IS Database | 32](#)

Confirm that the configuration is working properly.

## Checking Interface-to-Area Associations

### Purpose

Make sure that the interface-to-area associations are configured as expected.

### Action

From operational mode, enter the `show isis interface` command.

```
user@R5> show isis interface
IS-IS interface database:
Interface          L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0              3  0x1 Disabled           Passive         0/0
fe-1/2/0.0         2  0x3 Disabled          R5.03           10/10
fe-1/2/1.0         2  0x2 Disabled          R5.02           10/10
fe-1/2/2.0         1  0x1 R6.02             Disabled         10/10
fe-1/2/3.0         1  0x4 R5.04             Disabled         10/10
```

### Meaning

The output shows that Device R5's interfaces have been correctly configured with the ISO family, and that the interfaces have been placed into the correct levels.

You can also see that Device R5 has elected itself as the designated intermediate system (DIS) on its broadcast-capable IS-IS interfaces.

## Verifying IS-IS Adjacencies

### Purpose

Verify that the expected adjacencies have formed between Device R5 and its IS-IS neighbors.

### Action

From operational mode, enter the `show isis adjacency detail` command.

```
user@R5> show isis adjacency detail
R3
  Interface: fe-1/2/0.0, Level: 2, State: Up, Expires in 25 secs
```

Priority: 64, Up/Down transitions: 1, Last transition: 03:19:31 ago  
 Circuit type: 2, Speaks: IP, IPv6, MAC address: 0:5:85:8f:c8:bc  
 Topologies: Unicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R5.03, IP addresses: 10.0.0.21

#### R4

Interface: fe-1/2/1.0, Level: 2, State: Up, Expires in 24 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 03:19:36 ago  
 Circuit type: 2, Speaks: IP, IPv6, MAC address: 0:5:85:8f:c8:bc  
 Topologies: Unicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R5.02, IP addresses: 10.0.0.25

#### R6

Interface: fe-1/2/2.0, Level: 1, State: Up, Expires in 6 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 03:20:24 ago  
 Circuit type: 1, Speaks: IP, IPv6, MAC address: 0:5:85:8f:c8:bd  
 Topologies: Unicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R6.02, IP addresses: 10.0.0.30

#### R7

Interface: fe-1/2/3.0, Level: 1, State: Up, Expires in 21 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 03:19:29 ago  
 Circuit type: 1, Speaks: IP, IPv6, MAC address: 0:5:85:8f:c8:bc  
 Topologies: Unicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R5.04, IP addresses: 10.0.0.37

## Meaning

These results confirm that Device R5 has two Level 2 adjacencies and two Level 1 adjacencies.

## Examining the IS-IS Database

### Purpose

Because Device R5 is a Level 1/Level 2 (L1/L2) attached router, examine the Level 1 link-state database associated with area 49.0002 to confirm that loopback addresses from backbone routers are not being advertised into the Level 1 area.

## Action

From operational mode, enter the show isis database detail command.

```

user@R5> show isis database detail
IS-IS level 1 link-state database:

R5.00-00 Sequence: 0x19, Checksum: 0x7488, Lifetime: 727 secs
  IS neighbor: R5.04                      Metric:      10
  IS neighbor: R6.02                      Metric:      10
  IP prefix: 10.0.0.28/30                  Metric:      10 Internal Up
  IP prefix: 10.0.0.36/30                  Metric:      10 Internal Up

R5.04-00 Sequence: 0x14, Checksum: 0x2668, Lifetime: 821 secs
  IS neighbor: R5.00                      Metric:       0
  IS neighbor: R7.00                      Metric:       0

R6.00-00 Sequence: 0x17, Checksum: 0xa65, Lifetime: 774 secs
  IS neighbor: R6.02                      Metric:      10
  IS neighbor: R7.02                      Metric:      10
  IP prefix: 10.0.0.28/30                  Metric:      10 Internal Up
  IP prefix: 10.0.0.32/30                  Metric:      10 Internal Up
  IP prefix: 192.168.0.6/32                Metric:       0 Internal Up

R6.02-00 Sequence: 0x13, Checksum: 0xd1c0, Lifetime: 908 secs
  IS neighbor: R5.00                      Metric:       0
  IS neighbor: R6.00                      Metric:       0

R7.00-00 Sequence: 0x17, Checksum: 0xe39, Lifetime: 775 secs
  IS neighbor: R5.04                      Metric:      10
  IS neighbor: R7.02                      Metric:      10
  IP prefix: 10.0.0.32/30                  Metric:      10 Internal Up
  IP prefix: 10.0.0.36/30                  Metric:      10 Internal Up
  IP prefix: 192.168.0.7/32                Metric:       0 Internal Up

R7.02-00 Sequence: 0x13, Checksum: 0x404d, Lifetime: 966 secs
  IS neighbor: R6.00                      Metric:       0
  IS neighbor: R7.00                      Metric:       0

IS-IS level 2 link-state database:

R3.00-00 Sequence: 0x17, Checksum: 0x5f84, Lifetime: 1085 secs

```

IS neighbor: R4.02	Metric:	10
IS neighbor: R5.03	Metric:	10
IP prefix: 10.0.0.16/30	Metric:	10 Internal Up
IP prefix: 10.0.0.20/30	Metric:	10 Internal Up
IP prefix: 10.0.0.40/30	Metric:	10 Internal Up
IP prefix: 192.168.0.3/32	Metric:	0 Internal Up

R4.00-00 Sequence: 0x17, Checksum: 0xab3a, Lifetime: 949 secs

IS neighbor: R4.02	Metric:	10
IS neighbor: R5.02	Metric:	10
IP prefix: 10.0.0.16/30	Metric:	10 Internal Up
IP prefix: 10.0.0.24/30	Metric:	10 Internal Up
IP prefix: 192.168.0.4/32	Metric:	0 Internal Up

R4.02-00 Sequence: 0x14, Checksum: 0xf2a8, Lifetime: 1022 secs

IS neighbor: R3.00	Metric:	0
IS neighbor: R4.00	Metric:	0

R5.00-00 Sequence: 0x1f, Checksum: 0x20d7, Lifetime: 821 secs

IS neighbor: R5.02	Metric:	10
IS neighbor: R5.03	Metric:	10
IP prefix: 10.0.0.20/30	Metric:	10 Internal Up
IP prefix: 10.0.0.24/30	Metric:	10 Internal Up
IP prefix: 10.0.0.28/30	Metric:	10 Internal Up
IP prefix: 10.0.0.32/30	Metric:	20 Internal Up
IP prefix: 10.0.0.36/30	Metric:	10 Internal Up
IP prefix: 192.168.0.5/32	Metric:	0 Internal Up
IP prefix: 192.168.0.6/32	Metric:	10 Internal Up
IP prefix: 192.168.0.7/32	Metric:	10 Internal Up

R5.02-00 Sequence: 0x14, Checksum: 0x6135, Lifetime: 977 secs

IS neighbor: R4.00	Metric:	0
IS neighbor: R5.00	Metric:	0

R5.03-00 Sequence: 0x14, Checksum: 0x1483, Lifetime: 1091 secs

IS neighbor: R3.00	Metric:	0
IS neighbor: R5.00	Metric:	0



## Meaning

This display indicates that Device R5's loopback interface is correctly configured to run Level 2 only. Had Level 1 operation been enabled on lo0.0, Device R5 would have then included its loopback address in its Level 1 link-state PDU.

You can also see that Device R5 has Level 2 link-state PDUs, received from its adjacent neighbors.

Like an OSPF totally stubby area, no backbone (Level 2) or external prefixes are leaked into a Level 1 area, by default. Level 1 prefixes are leaked up into the IS-IS backbone, however, as can be seen in Device R5's Level 2 link-state PDU.

## RELATED DOCUMENTATION

[Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups](#) | 21

## Understanding IS-IS Designated Routers

A router advertises its priority to become a designated router in its hello packets. On all multiaccess networks (physical networks that support the attachment of more than two routers, such as Ethernet networks), IS-IS uses the advertised priorities to elect a designated router for the network. This router is responsible for sending network link-state advertisements, which describe all the routers attached to the network. These advertisements are flooded throughout a single area. The priority value is meaningful only on a multiaccess network. It has no meaning on a point-to-point interface.

A router's priority for becoming the designated router is indicated by an arbitrary number from 0 through 127, which you configure on the IS-IS interface. The router with the highest priority becomes the designated router for the area (Level 1, Level 2, or both), also configured on the IS-IS interface. If routers in the network have the same priority, then the router with the highest MAC address is elected as the designated router. By default, routers have a priority value of 64.

In IS-IS protocol, a pseudo LSP (Link State PDU) is used to represent a pseudo node, which is a logical node that is not a real physical device. The designated router becomes a pseudonode and advertises pseudonode LSPs. IS-IS modifies the deployment as a hub-and-spoke topology with all nodes in the same broadcast domain having a point-to-point connection with a pseudonode. This helps in avoiding a full-mesh of point-to-point connections between all nodes in the same LAN segment and simplifies the network topology.

Each LSP is identified by its <name>.<pseudonodeid>-<fragment-id>. You can distinguish between a regular LSP and a pseudo LSP by checking the <pseudonodeid> value. A regular LSP has a <pseudonodeid> value of zero, whereas a pseudo LSP has a non-zero value in its LSP packet.

## RELATED DOCUMENTATION

[Configuring Designated Router Election Priority for IS-IS | 36](#)

## Configuring Designated Router Election Priority for IS-IS

This example shows how to configure the designated router election priority for IS-IS.

Before you begin:

- Configure network interfaces. See the *Junos OS Interfaces Configuration Guide for Security Devices*.
- Enable IS-IS on the interfaces. See ["Example: Configuring IS-IS" on page 15](#).

In this example, you configure the priority for logical interface ge-0/0/1.0 to be 100 and the level number to be 1. If this interface has the highest priority value, the router becomes the designated router for the Level 1 area.

To configure a designated router election priority for IS-IS:

[edit]

```
user@host# set protocols isis interface ge-0/0/1.0 level 1 priority 100
```



### NOTE:

- The default designated router election priority for IS-IS is 64. If you do not configure the priority value for an interface, by default IS-IS assigns the priority 64.
- The router with the highest priority value is elected as the designated router.

## RELATED DOCUMENTATION

[Understanding IS-IS Designated Routers | 35](#)

[Example: Configuring IS-IS | 15](#)

## Configuring an ISO System Identifier for the Router

For IS-IS to operate on the router, you can optionally configure a system identifier (system ID). The system identifier is commonly the media access control (MAC) address or the IP address expressed in binary-coded decimal (BCD).

If you do not statically map the hostname, the mapping is generated dynamically, based on the system host-name. If you omit the `static-host-mapping hostname sysid` statement, the IS-IS system ID is dynamically generated from the host portion of the ISO address configured on the loopback interface (lo0) and is mapped to the `hostname` statement configured at the `[edit system]` hierarchy level. Run the `show isis hostname` command to view the mappings.

To configure an International Organization for Standardization (ISO) system ID, include the `sysid` statement at the `[edit system static-host-mapping hostname]` hierarchy level:

```
[edit system]
static-host-mapping {
  hostname {
    sysid system-identifier;
  }
}
```

*hostname* is the name specified by the `host-name` statement at the `[edit system]` hierarchy level.

*system-identifier* is the ISO system identifier. It is the 6-byte system ID portion of the IS-IS network service access point (NSAP). We recommend that you use the host's IP address represented in BCD format. For example, the IP address 192.168.1.77 is 1921.6800.1077 in BCD.

### RELATED DOCUMENTATION

*Configuring a Device's Unique Identity for the Network*

## Understanding Default Routes

A default route is the route that takes effect when no other route is available for an IP destination address.

If a packet is received on a routing device, the device first checks to see if the IP destination address is on one of the device's local subnets. If the destination address is not local, the device checks its routing table. If the remote destination subnet is not listed in the routing table, the packet is forwarded to the

next hop toward the destination using the default route. The default route generally has a next-hop address of another routing device, which performs the same process. The process repeats until a packet is delivered to the destination.

The route evaluation process in each router uses the longest prefix match method to obtain the most specific route. The network with the longest subnet mask that matches the destination IP address is the next-hop network gateway.

The default route in IPv4 is designated as 0.0.0.0/0 or simply 0/0. Similarly, in IPv6, the default route is specified as ::/0. The subnet mask /0 specifies all networks, and is the shortest match possible. A route lookup that does not match any other route uses this route if it is configured and active in the routing table. To be active, the configured next-hop address must be reachable.

Administrators generally point the default route toward the routing device that has a connection to a network service provider. Therefore, packets with destinations outside the organization's local area network, typically destinations on the Internet or a wide area network, are forwarded to the routing device with the connection to that provider. The device to which the default route points is often called the default gateway.

## RELATED DOCUMENTATION

[Example: Configuring an IS-IS Default Route Policy on Logical Systems | 736](#)

*Example: Configuring an OSPF Default Route Policy on Logical Systems*

*Example: Configuring a Conditional Default Route Policy*

## How to Configure Multiple Independent IGP Instances of IS-IS and OSPFv2

### SUMMARY

Learn how to configure and run multiple instances of IGP on a router.

### IN THIS SECTION

- [Configure Multiple IGP Instances of IS-IS | 39](#)
- [Example: Configure Independent IS-IS Instances in Metro Flooding Domains | 41](#)

- [Example: Configure Multiple Independent Instances of OSPFv2 with Segment Routing | 67](#)

## Configure Multiple IGP Instances of IS-IS

### SUMMARY

Learn about the benefits and get an overview of running multiple interior gateway protocol (IGP) instances of IS-IS on a router.

### IN THIS SECTION

- [Benefits of Multi-Instance IS-IS | 39](#)
- [Multi-Instance IS-IS Overview | 40](#)

### Benefits of Multi-Instance IS-IS

- You can use multiple IGP instances of IS-IS to redistribute routes among independent IS-IS domains on a single router.
- You can construct flexible IS-IS hierarchies across independent IGP domains.
- Allows decoupling of multiple IS-IS flooding domains and therefore achieve a more scalable IS-IS deployment.

Figure 5: Multi-Instance IS-IS Deployment Topology

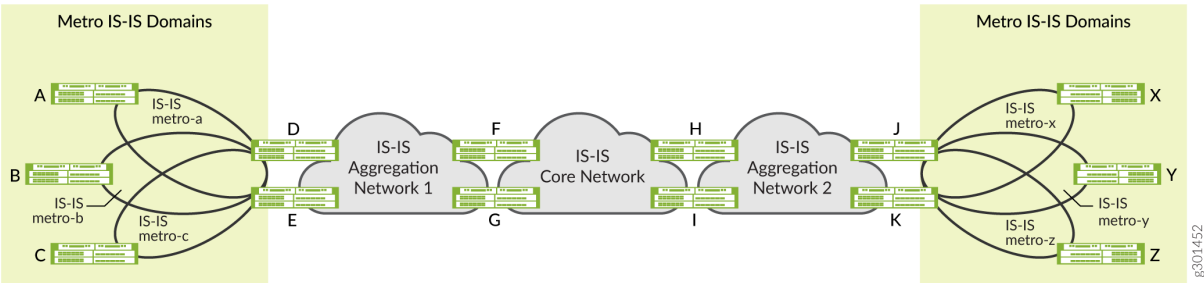


Figure 5 on page 39 illustrates several benefits of configuring multiple IGP instances of IS-IS on the router. For example, Router F participates in two independent IS-IS instances. Router F treats IS-IS

Aggregation Network-1 and IS-IS Core Network as two independent IGP domains, while at the same time redistributing routes between those domains. Network operators can use this flexibility to construct a hierarchy of IS-IS domains.

Figure 5 on page 39 also illustrates the use of multiple IGP instances of IS-IS to separate metro networks into independent IS-IS flooding domains. In the example, routers D and E participate in the IS-IS metro-a, IS-IS metro-b, and IS-IS metro-c networks, as well as in IS-IS Aggregation Network-1. Routers D and E do not flood the different IS-IS domains with IS-IS advertisements. Instead they redistribute specific routes among the different IS-IS domains, which allows for more scalable metro deployments.

### Multi-Instance IS-IS Overview

You can configure and run multiple independent IGP instances of IS-IS simultaneously on a router. These instances are associated with the default routing instance, and they install routes in the default routing table. Each IS-IS instance can also export the routes installed in the routing table by other IS-IS instances using the standard Junos OS routing policy configuration. By default, the routes installed by the different IS-IS instances have the same route preference.



**NOTE:** Junos OS does not support configuring the same logical interface in multiple IGP instances of IS-IS.

In most deployment scenarios, only one IS-IS instance on a router installs a route for a given prefix. Therefore, you don't need to configure different route preferences for multiple IS-IS instances. However, for certain deployment scenarios where multiple IS-IS instances install the routes for the same prefix in the routing table, you can set a different route preference for the routes installed by other IS-IS instances. This allows the routing table to choose the routes with the best route preference and installs those routes in the forwarding table.

You can use the multiple IS-IS instance feature for both hierarchical and parallel deployments. In the case of hierarchical deployments, there are well-defined borders between the groups of routers participating in different IGP instances. In parallel deployments, different IGP instances (typically not more than two or three) span entire groups of routers. You can also have mixed deployments, with some domains in a hierarchical deployment running IGP instances in parallel.

You can configure multiple independent IGP instances of IS-IS by including the `isis-instance` configuration statement at the `[edit protocols]` hierarchy level. The configuration statements that you use at the `[edit protocols isis-instance igp-instance-name]` hierarchy level are the same as those available at the `[edit protocols isis]` hierarchy level.



**NOTE:** The `isis-instance` configuration statement is not supported at the `[edit routing-instances routing-instance-name protocols]` hierarchy level.

You can configure multiple independent IGP instances of OSPFv2 by including the `ospf-instance` configuration statement at the `[edit protocols]` hierarchy level. The configuration statements that you use at the `[edit protocols ospf-instance igp-instance-name]` hierarchy level are the same as those available at the `[edit protocols ospf]` hierarchy level.



**NOTE:** The `ospf-instance` configuration statement is not supported at the `[edit routing-instances routing-instance-name protocols]` hierarchy level.

You can configure and run multiple independent interior gateway protocol (IGP) instances of OSPFv2 with segment routing (SR) on a router. You can create two or more OSPF instances and apply SR-MPLS on each instance. Multiple instances of OSPF can advertise different prefix-segment identifiers (prefix-SIDs). Other instances can use these SIDs for making routing decisions.

Multi-instance OSPF combined with SR enhances network flexibility, scalability, and control over traffic engineering, especially in large and complex networks.

## Example: Configure Independent IS-IS Instances in Metro Flooding Domains

### SUMMARY

Use this example to learn how to configure independent metro flooding domains running multiple IGP instances of IS-IS.

### IN THIS SECTION

- [Overview | 41](#)
- [Requirements | 42](#)
- [Configuration | 43](#)
- [Verification | 60](#)

### Overview

#### IN THIS SECTION

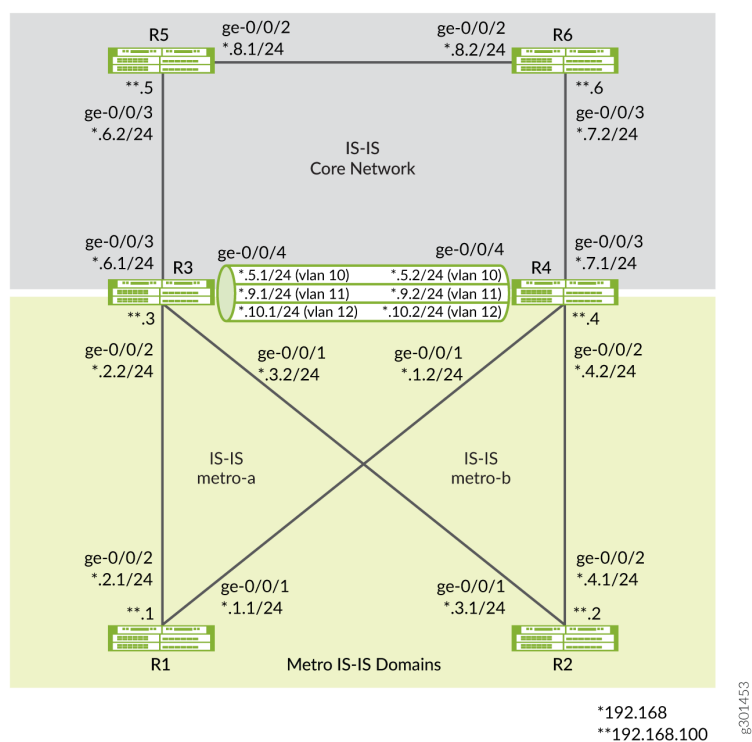
- [Topology | 42](#)

This example shows how to configure and run multiple independent IGP instances of IS-IS in metro flooding domains.

### Topology

Figure 2 shows an example of metro flooding domains (metro-a and metro-b) running independent IGP instances of IS-IS. In the topology, routers R3 and R4 participate in metro IS-IS domains (IS-IS metro-a and IS-IS metro-b) and the IS-IS core network domain. Routers R3 and R4 do not flood the different IS-IS domains with IS-IS advertisements. Instead they redistribute specific routes among the different IS-IS domains, which allows for a more scalable metro deployment.

**Figure 6: Multi-Instance IS-IS Topology Across Independent Metro Flooding Domains (IGP Domains)**



### Requirements

This example uses the following hardware and software components:

- MX Series routers
- Junos OS Release 21.1R1 or later running on all devices





**NOTE:** You must configure the network services mode as Enhanced IP. The Enhanced IP configuration ensures that the router uses enhanced mode capabilities.

[edit]

```
user@CE1#set chassis network-services enhanced-ip
```

After you configure the enhanced-ip statement and commit the configuration, the following warning message appears, prompting you to reboot the router:

```
'chassis'
```

```
WARNING: Chassis configuration for network services has been changed. A system reboot
is mandatory. Please reboot the system NOW. Continuing without a reboot might result
in unexpected system behavior.
```

```
commit complete
```

The reboot brings up the FPCs on the router.

[See [show chassis network-services.](#)]

## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 43](#)
- [Configure R1 | 49](#)
- [Configure R3 | 52](#)

To configure and run multiple IGP instances of IS-IS on the router, perform these tasks:

### *CLI Quick Configuration*

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

## Device R1

```

set interfaces ge-0/0/1 description R1-to-R4
set interfaces ge-0/0/1 unit 0 family inet address 192.168.1.1/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/2 description R1-to-R3
set interfaces ge-0/0/2 unit 0 family inet address 192.168.2.1/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.100.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set protocols isis interface ge-0/0/1.0 level 2 metric 100
set protocols isis interface ge-0/0/1.0 level 1 disable
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/2.0 level 1 disable
set protocols isis interface ge-0/0/2.0 level 2 metric 100
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface lo0.0 passive
set routing-options router-id 192.168.100.1

```

## Device R2

```

set interfaces ge-0/0/1 description R2-to-R3
set interfaces ge-0/0/1 unit 0 family inet address 192.168.3.1/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/2 description R2-to-R4
set interfaces ge-0/0/2 unit 0 family inet address 192.168.4.1/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.100.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0002.00
set protocols isis interface ge-0/0/1.0 level 1 disable
set protocols isis interface ge-0/0/1.0 level 2 metric 100
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/2.0 level 1 disable
set protocols isis interface ge-0/0/2.0 level 2 metric 100
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface lo0.0 passive
set routing-options router-id 192.168.100.2

```

## Device R3

```

set interfaces ge-0/0/1 description R3-to-R2
set interfaces ge-0/0/1 unit 0 family inet address 192.168.3.2/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/2 description R3-to-R1
set interfaces ge-0/0/2 unit 0 family inet address 192.168.2.2/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/3 description R3-to-R5
set interfaces ge-0/0/3 unit 0 family inet address 192.168.6.1/24
set interfaces ge-0/0/3 unit 0 family iso
set interfaces ge-0/0/4 description R3-to-R4
set interfaces ge-0/0/4 vlan-tagging
set interfaces ge-0/0/4 unit 0 vlan-id 10
set interfaces ge-0/0/4 unit 0 family inet address 192.168.5.1/24
set interfaces ge-0/0/4 unit 0 family iso
set interfaces ge-0/0/4 unit 1 vlan-id 11
set interfaces ge-0/0/4 unit 1 family inet address 192.168.9.1/24
set interfaces ge-0/0/4 unit 1 family iso
set interfaces ge-0/0/4 unit 2 vlan-id 12
set interfaces ge-0/0/4 unit 2 family inet address 192.168.10.1/24
set interfaces ge-0/0/4 unit 2 family iso
set interfaces lo0 unit 0 family inet address 192.168.100.3/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0003.00
set policy-options policy-statement export-direct-loopback from protocol direct
set policy-options policy-statement export-direct-loopback from route-filter 192.168.100.3/32
exact
set policy-options policy-statement export-direct-loopback then accept
set policy-options policy-statement export-isis from protocol isis
set policy-options policy-statement export-isis from level 2
set policy-options policy-statement export-isis from route-filter 192.168.100.0/24 longer
set policy-options policy-statement export-isis then accept
set policy-options policy-statement export-isis-metro-a from igp-instance metro-a
set policy-options policy-statement export-isis-metro-a from protocol isis
set policy-options policy-statement export-isis-metro-a from level 2
set policy-options policy-statement export-isis-metro-a from route-filter 192.168.100.0/24 longer
set policy-options policy-statement export-isis-metro-a then accept
set policy-options policy-statement export-isis-metro-b from igp-instance metro-b
set policy-options policy-statement export-isis-metro-b from protocol isis
set policy-options policy-statement export-isis-metro-b from level 2
set policy-options policy-statement export-isis-metro-b from route-filter 192.168.100.0/24 longer

```

```

set policy-options policy-statement export-isis-metro-b then accept
set protocols isis interface ge-0/0/3.0 level 1 disable
set protocols isis interface ge-0/0/3.0 level 2 metric 100
set protocols isis interface ge-0/0/3.0 point-to-point
set protocols isis interface ge-0/0/4.0 level 1 disable
set protocols isis interface ge-0/0/4.0 level 2 metric 100
set protocols isis interface ge-0/0/4.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis export export-isis-metro-a
set protocols isis export export-isis-metro-b
set protocols isis-instance metro-b interface ge-0/0/1.0 level 1 disable
set protocols isis-instance metro-b interface ge-0/0/1.0 level 2 metric 100
set protocols isis-instance metro-b interface ge-0/0/1.0 point-to-point
set protocols isis-instance metro-b interface ge-0/0/4.2 level 1 disable
set protocols isis-instance metro-b interface ge-0/0/4.2 level 2 metric 100
set protocols isis-instance metro-b interface ge-0/0/4.2 point-to-point
set protocols isis-instance metro-b export export-isis
set protocols isis-instance metro-b export export-direct-loopback
set protocols isis-instance metro-b export export-isis-metro-a
set protocols isis-instance metro-a interface ge-0/0/2.0 level 1 disable
set protocols isis-instance metro-a interface ge-0/0/2.0 level 2 metric 100
set protocols isis-instance metro-a interface ge-0/0/2.0 point-to-point
set protocols isis-instance metro-a interface ge-0/0/4.1 level 1 disable
set protocols isis-instance metro-a interface ge-0/0/4.1 level 2 metric 100
set protocols isis-instance metro-a interface ge-0/0/4.1 point-to-point
set protocols isis-instance metro-a export export-isis
set protocols isis-instance metro-a export export-direct-loopback
set protocols isis-instance metro-a export export-isis-metro-b
set routing-options router-id 192.168.100.3

```

## Device R4

```

set interfaces ge-0/0/1 description R4-to-R1
set interfaces ge-0/0/1 unit 0 family inet address 192.168.1.2/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/2 description R4-to-R2
set interfaces ge-0/0/2 unit 0 family inet address 192.168.4.2/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/3 description R4-to-R6
set interfaces ge-0/0/3 unit 0 family inet address 192.168.7.1/24
set interfaces ge-0/0/3 unit 0 family iso

```

```

set interfaces ge-0/0/4 description R4-to-R3
set interfaces ge-0/0/4 vlan-tagging
set interfaces ge-0/0/4 unit 0 vlan-id 10
set interfaces ge-0/0/4 unit 0 family inet address 192.168.5.2/24
set interfaces ge-0/0/4 unit 0 family iso
set interfaces ge-0/0/4 unit 1 vlan-id 11
set interfaces ge-0/0/4 unit 1 family inet address 192.168.9.2/24
set interfaces ge-0/0/4 unit 1 family iso
set interfaces ge-0/0/4 unit 2 vlan-id 12
set interfaces ge-0/0/4 unit 2 family inet address 192.168.10.2/24
set interfaces ge-0/0/4 unit 2 family iso
set interfaces lo0 unit 0 family inet address 192.168.100.4/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0004.00
set policy-options policy-statement export-direct-loopback from protocol direct
set policy-options policy-statement export-direct-loopback from route-filter 192.168.100.4/32
exact
set policy-options policy-statement export-direct-loopback then accept
set policy-options policy-statement export-isis from protocol isis
set policy-options policy-statement export-isis from level 2
set policy-options policy-statement export-isis from route-filter 192.168.100.0/24 longer
set policy-options policy-statement export-isis then accept
set policy-options policy-statement export-isis-metro-a from igp-instance metro-a
set policy-options policy-statement export-isis-metro-a from protocol isis
set policy-options policy-statement export-isis-metro-a from level 2
set policy-options policy-statement export-isis-metro-a from route-filter 192.168.100.0/24 longer
set policy-options policy-statement export-isis-metro-a then accept
set policy-options policy-statement export-isis-metro-b from igp-instance metro-b
set policy-options policy-statement export-isis-metro-b from protocol isis
set policy-options policy-statement export-isis-metro-b from level 2
set policy-options policy-statement export-isis-metro-b from route-filter 192.168.100.0/24 longer
set policy-options policy-statement export-isis-metro-b then accept
set protocols isis interface ge-0/0/3.0 level 1 disable
set protocols isis interface ge-0/0/3.0 level 2 metric 100
set protocols isis interface ge-0/0/3.0 point-to-point
set protocols isis interface ge-0/0/4.0 level 1 disable
set protocols isis interface ge-0/0/4.0 level 2 metric 100
set protocols isis interface ge-0/0/4.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis export export-isis-metro-a
set protocols isis export export-isis-metro-b
set protocols isis-instance metro-a interface ge-0/0/1.0 level 1 disable
set protocols isis-instance metro-a interface ge-0/0/1.0 level 2 metric 100
set protocols isis-instance metro-a interface ge-0/0/1.0 point-to-point

```

```

set protocols isis-instance metro-a interface ge-0/0/4.1 level 1 disable
set protocols isis-instance metro-a interface ge-0/0/4.1 level 2 metric 100
set protocols isis-instance metro-a interface ge-0/0/4.1 point-to-point
set protocols isis-instance metro-a export export-isis
set protocols isis-instance metro-a export export-direct-loopback
set protocols isis-instance metro-a export export-isis-metro-b
set protocols isis-instance metro-b interface ge-0/0/2.0 level 1 disable
set protocols isis-instance metro-b interface ge-0/0/2.0 level 2 metric 100
set protocols isis-instance metro-b interface ge-0/0/2.0 point-to-point
set protocols isis-instance metro-b interface ge-0/0/4.2 level 1 disable
set protocols isis-instance metro-b interface ge-0/0/4.2 level 2 metric 100
set protocols isis-instance metro-b interface ge-0/0/4.2 point-to-point
set protocols isis-instance metro-b export export-isis
set protocols isis-instance metro-b export export-direct-loopback
set protocols isis-instance metro-b export export-isis-metro-a
set routing-options router-id 192.168.100.4

```

## Device R5

```

set interfaces ge-0/0/2 description R5-to-R6
set interfaces ge-0/0/2 unit 0 family inet address 192.168.8.1/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/3 description R5-to-R3
set interfaces ge-0/0/3 unit 0 family inet address 192.168.6.2/24
set interfaces ge-0/0/3 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.100.5/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0005.00
set protocols isis interface ge-0/0/2.0 level 1 disable
set protocols isis interface ge-0/0/2.0 level 2 metric 100
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface ge-0/0/3.0 level 1 disable
set protocols isis interface ge-0/0/3.0 level 2 metric 100
set protocols isis interface ge-0/0/3.0 point-to-point
set protocols isis interface lo0.0 passive
set routing-options router-id 192.168.100.5

```

## Device R6

```

set interfaces ge-0/0/2 description R6-to-R5
set interfaces ge-0/0/2 unit 0 family inet address 192.168.8.2/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/3 description R6-to-R4
set interfaces ge-0/0/3 unit 0 family inet address 192.168.7.2/24
set interfaces ge-0/0/3 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.100.6/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0006.00
set protocols isis interface ge-0/0/2.0 level 1 disable
set protocols isis interface ge-0/0/2.0 level 2 metric 100
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface ge-0/0/3.0 level 1 disable
set protocols isis interface ge-0/0/3.0 level 2 metric 100
set protocols isis interface ge-0/0/3.0 point-to-point
set protocols isis interface lo0.0 passive
set routing-options router-id 192.168.100.6

```

## Configure R1

### Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see [Using the CLI Editor in Configuration Mode](#) in the [CLI User Guide](#).

You can use the steps in this example to also configure the R2, R5, and R6 routers. See ["CLI Quick Configuration" on page 43](#) and Figure 2 to understand the interface IDs, IP addresses, and the loopback addresses used on these routers.

To configure R1:

1. Configure the interfaces to enable IP (inet) and ISO family support.

```

user@R1# set interfaces ge-0/0/1 description R1-to-R4
user@R1# set interfaces ge-0/0/1 unit 0 family inet address 192.168.1.1/24
user@R1# set interfaces ge-0/0/1 unit 0 family iso
user@R1# set interfaces ge-0/0/2 description R1-to-R3

```

```
user@R1# set interfaces ge-0/0/2 unit 0 family inet address 192.168.2.1/24
user@R1# set interfaces ge-0/0/2 unit 0 family iso
```

2. Create the loopback interface and configure the IP and NET addresses.

```
user@R1# set interfaces lo0 unit 0 family inet address 192.168.100.1/32
user@R1# set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
```

3. Configure routing options to identify the router in the domain.

```
user@R1# set routing-options router-id 192.168.100.1
```

4. Enable IS-IS on the interfaces.

```
user@R1# set protocols isis interface ge-0/0/1.0 level 2 metric 100
user@R1# set protocols isis interface ge-0/0/1.0 level 1 disable
user@R1# set protocols isis interface ge-0/0/1.0 point-to-point
user@R1# set protocols isis interface ge-0/0/2.0 level 1 disable
user@R1# set protocols isis interface ge-0/0/2.0 level 2 metric 100
user@R1# set protocols isis interface ge-0/0/2.0 point-to-point
user@R1# set protocols isis interface lo0.0 passive
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show routing-options`, and `show protocols` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
interfaces {
  ge-0/0/1 {
    description R1-to-R4;
    unit 0 {
      family inet {
        address 192.168.1.1/24;
```



```

        }
        family iso;
    }
}
ge-0/0/2 {
    description R1-to-R3;
    unit 0 {
        family inet {
            address 192.168.2.1/24;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.100.1/32;
        }
        family iso {
            address 49.0002.0192.0168.0001.00;
        }
    }
}
}
protocols {
    isis {
        interface ge-0/0/1.0 {
            level 2 metric 100;
            level 1 disable;
            point-to-point;
        }
        interface ge-0/0/2.0 {
            level 1 disable;
            level 2 metric 100;
            point-to-point;
        }
        interface lo0.0 {
            passive;
        }
    }
}
routing-options {

```

```
router-id 192.168.100.1;
}
```

### Configure R3

#### Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see [Using the CLI Editor in Configuration Mode](#) in the [CLI User Guide](#).

You can use the steps in this example to also configure the R4 router. See ["CLI Quick Configuration" on page 43](#) and Figure 2 to understand the interface IDs, IP addresses, and the loopback address used on the router.

To configure R3:

1. Configure the interfaces connecting to R1, R2, and R5 to enable IP and ISO family support.

```
user@R3# set interfaces ge-0/0/1 description R3-to-R2
user@R3# set interfaces ge-0/0/1 unit 0 family inet address 192.168.3.2/24
user@R3# set interfaces ge-0/0/1 unit 0 family iso
user@R3# set interfaces ge-0/0/2 description R3-to-R1
user@R3# set interfaces ge-0/0/2 unit 0 family inet address 192.168.2.2/24
user@R3# set interfaces ge-0/0/2 unit 0 family iso
user@R3# set interfaces ge-0/0/3 description R3-to-R5
user@R3# set interfaces ge-0/0/3 unit 0 family inet address 192.168.6.1/24
user@R3# set interfaces ge-0/0/3 unit 0 family iso
```

2. Configure three subinterfaces (logical interfaces) connecting R3 and R4 (one IS-IS standard instance and two IS-IS metro instances (IS-IS metro-a and IS-IS metro-b)).



**NOTE:** The standard IS-IS instance refers to the IS-IS IGP instance configured at the [edit protocols isis] hierarchy level.

```
user@R3# set interfaces ge-0/0/4 description R3-to-R4
user@R3# set interfaces ge-0/0/4 vlan-tagging
user@R3# set interfaces ge-0/0/4 unit 0 vlan-id 10
```

```

user@R3# set interfaces ge-0/0/4 unit 0 family inet address 192.168.5.1/24
user@R3# set interfaces ge-0/0/4 unit 0 family iso
user@R3# set interfaces ge-0/0/4 unit 1 vlan-id 11
user@R3# set interfaces ge-0/0/4 unit 1 family inet address 192.168.9.1/24
user@R3# set interfaces ge-0/0/4 unit 1 family iso
user@R3# set interfaces ge-0/0/4 unit 2 vlan-id 12
user@R3# set interfaces ge-0/0/4 unit 2 family inet address 192.168.10.1/24
user@R3# set interfaces ge-0/0/4 unit 2 family iso

```

3. Create the loopback interface and configure the IP and NET addresses.

```

user@R3# set interfaces lo0 unit 0 family inet address 192.168.100.3/32
user@R3# set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0003.00

```

4. Configure policies to redistribute loopback addresses of IS-IS metro-instance (IS-IS metro-a and IS-IS metro-b) and IS-IS standard-instance (core network) routers, so that the routes can be distributed across IS-IS domains as required.

- a. Configure policies to distribute the loopback address of R3.

```

user@R3# set policy-options policy-statement export-direct-loopback from protocol direct
user@R3# set policy-options policy-statement export-direct-loopback from route-filter
192.168.100.3/32 exact
user@R3# set policy-options policy-statement export-direct-loopback then accept

```

- b. Configure policies to distribute the loopback addresses of the R5 and R6 routers (standard IS-IS instance).

```

user@R3# set policy-options policy-statement export-isis from protocol isis
user@R3# set policy-options policy-statement export-isis from level 2
user@R3# set policy-options policy-statement export-isis from route-filter
192.168.100.0/24 longer
user@R3# set policy-options policy-statement export-isis then accept

```

- c. Configure policies to distribute the loopback addresses of R1 (IS-IS metro-a instance).

```

user@R3# set policy-options policy-statement export-isis-metro-a from igp-instance metro-
a
user@R3# set policy-options policy-statement export-isis-metro-a from protocol isis
user@R3# set policy-options policy-statement export-isis-metro-a from level 2
user@R3# set policy-options policy-statement export-isis-metro-a from route-filter
192.168.100.0/24 longer
user@R3# set policy-options policy-statement export-isis-metro-a then accept

```

- d. Configure policies to distribute the loopback addresses of R2 (IS-IS metro-b instance).

```

user@R3# set policy-options policy-statement export-isis-metro-b from igp-instance metro-
b
user@R3# set policy-options policy-statement export-isis-metro-b from protocol isis
user@R3# set policy-options policy-statement export-isis-metro-b from level 2
user@R3# set policy-options policy-statement export-isis-metro-b from route-filter
192.168.100.0/24 longer
user@R3# set policy-options policy-statement export-isis-metro-b then accept

```

5. Enable IS-IS on the standard-instance interface (connecting R3 to R5) and on the subinterface (connecting R3 to R4).

```

user@R3# set protocols isis interface ge-0/0/3.0 level 1 disable
user@R3# set protocols isis interface ge-0/0/3.0 level 2 metric 100
user@R3# set protocols isis interface ge-0/0/3.0 point-to-point
user@R3# set protocols isis interface ge-0/0/4.0 level 1 disable
user@R3# set protocols isis interface ge-0/0/4.0 level 2 metric 100
user@R3# set protocols isis interface ge-0/0/4.0 point-to-point
user@R3# set protocols isis interface lo0.0 passive

```

6. Configure IS-IS to export loopback addresses from IS-IS metro-a and IS-IS metro-b instances to the IS-IS standard instance. This configuration distributes specific routes instead of flooding the entire metro domain.

```
user@R3# set protocols isis export export-isis-metro-a
user@R3# set protocols isis export export-isis-metro-b
```

7. Enable IS-IS on the IS-IS metro-b instance interface (connecting R3 to R2) and on the subinterface (R3 to R4).

```
user@R3# set protocols isis-instance metro-b interface ge-0/0/1.0 level 1 disable
user@R3# set protocols isis-instance metro-b interface ge-0/0/1.0 level 2 metric 100
user@R3# set protocols isis-instance metro-b interface ge-0/0/1.0 point-to-point
user@R3# set protocols isis-instance metro-b interface ge-0/0/4.2 level 1 disable
user@R3# set protocols isis-instance metro-b interface ge-0/0/4.2 level 2 metric 100
user@R3# set protocols isis-instance metro-b interface ge-0/0/4.2 point-to-point
```

8. Configure IS-IS to export the loopback addresses of IS-IS metro-a and standard IS-IS instances to the IS-IS metro-b instance. This configuration distributes specific routes instead of flooding the entire standard IS-IS instances and metro-a domain instances.

```
user@R3# set protocols isis-instance metro-b export export-isis
user@R3# set protocols isis-instance metro-b export export-direct-loopback
user@R3# set protocols isis-instance metro-b export export-isis-metro-a
```

9. Enable IS-IS on the IS-IS metro-a instance interface (connecting R3 to R1) and on the subinterface (R3 to R4).

```
user@R3# set protocols isis-instance metro-a interface ge-0/0/2.0 level 1 disable
user@R3# set protocols isis-instance metro-a interface ge-0/0/2.0 level 2 metric 100
user@R3# set protocols isis-instance metro-a interface ge-0/0/2.0 point-to-point
user@R3# set protocols isis-instance metro-a interface ge-0/0/4.1 level 1 disable
user@R3# set protocols isis-instance metro-a interface ge-0/0/4.1 level 2 metric 100
user@R3# set protocols isis-instance metro-a interface ge-0/0/4.1 point-to-point
```

10. Configure IS-IS to export the loopback addresses of IS-IS metro-b and standard IS-IS instances to the IS-IS metro-a instance. This configuration distributes specific routes instead of flooding the entire standard IS-IS instances and metro-b domain instances.

```
user@R3# set protocols isis-instance metro-a export export-isis
user@R3# set protocols isis-instance metro-a export export-direct-loopback
user@R3# set protocols isis-instance metro-a export export-isis-metro-b
```

11. Configure routing options to identify the router in the domain.

```
user@R3# set routing-options router-id 192.168.100.3
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show routing-options`, and `show protocols` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
interfaces {
  ge-0/0/1 {
    description R3-to-R2;
    unit 0 {
      family inet {
        address 192.168.3.2/24;
      }
      family iso;
    }
  }
  ge-0/0/2 {
    description R3-to-R1;
    unit 0 {
      family inet {
        address 192.168.2.2/24;
      }
      family iso;
    }
  }
}
```

```

}
ge-0/0/3 {
    description R3-to-R5;
    unit 0 {
        family inet {
            address 192.168.6.1/24;
        }
        family iso;
    }
}
ge-0/0/4 {
    description R3-to-R4;
    vlan-tagging;
    unit 0 {
        vlan-id 10;
        family inet {
            address 192.168.5.1/24;
        }
        family iso;
    }
    unit 1 {
        vlan-id 11;
        family inet {
            address 192.168.9.1/24;
        }
        family iso;
    }
    unit 2 {
        vlan-id 12;
        family inet {
            address 192.168.10.1/24;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.100.3/32;
        }
        family iso {
            address 49.0002.0192.0168.0003.00;
        }
    }
}

```

```

    }
  }
}
policy-options {
  policy-statement export-direct-loopback {
    from {
      protocol direct;
      route-filter 192.168.100.3/32 exact;
    }
    then accept;
  }
  policy-statement export-isis {
    from {
      protocol isis;
      level 2;
      route-filter 192.168.100.0/24 longer;
    }
    then accept;
  }
  policy-statement export-isis-metro-a {
    from {
      igp-instance metro-a;
      protocol isis;
      level 2;
      route-filter 192.168.100.0/24 longer;
    }
    then accept;
  }
  policy-statement export-isis-metro-b {
    from {
      igp-instance metro-b;
      protocol isis;
      level 2;
      route-filter 192.168.100.0/24 longer;
    }
    then accept;
  }
}
protocols {
  isis {
    interface ge-0/0/3.0 {
      level 1 disable;
      level 2 metric 100;
    }
  }
}

```



```

        point-to-point;
    }
    interface ge-0/0/4.0 {
        level 1 disable;
        level 2 metric 100;
        point-to-point;
    }
    interface lo0.0 {
        passive;
    }
    export [ export-isis-metro-a export-isis-metro-b ];
}
isis-instance metro-b {
    interface ge-0/0/1.0 {
        level 1 disable;
        level 2 metric 100;
        point-to-point;
    }
    interface ge-0/0/4.2 {
        level 1 disable;
        level 2 metric 100;
        point-to-point;
    }
    export [ export-isis export-direct-loopback export-isis-metro-a ];
}
isis-instance metro-a {
    interface ge-0/0/2.0 {
        level 1 disable;
        level 2 metric 100;
        point-to-point;
    }
    interface ge-0/0/4.1 {
        level 1 disable;
        level 2 metric 100;
        point-to-point;
    }
    export [ export-isis export-direct-loopback export-isis-metro-b ];
}
}
routing-options {
    router-id 192.168.100.3;
}

```

## Verification

### IN THIS SECTION

- [Verify IS-IS Advertisements | 60](#)
- [Verify the Routing Table | 62](#)
- [Verify the Routes in the IS-IS Routing Table | 63](#)
- [Verify IS-IS Interfaces | 65](#)

To verify that the configuration is working properly, perform the following tasks:

### *Verify IS-IS Advertisements*

#### Purpose

Verify the IS-IS advertisement entries in the IS-IS link-state database (LSDB), which contains data about PDU packets.

#### Action

From operational mode, run the `show isis database level 2` command.

#### On R3

```
user@R3>show isis database level 2
```

IS-IS level 2 link-state database:

LSP ID	Sequence	Checksum	Lifetime	Attributes
R6.00-00	0x75d	0x1ff7	1181	L1 L2
R5.00-00	0x75b	0xffdc	741	L1 L2
R4.00-00	0x780	0x4e1	552	L1 L2
R3.00-00	0x7f0	0x8643	496	L1 L2
4 LSPs				

```
user@R3>show isis database level 2 igp-instance metro-a
```

IS-IS level 2 link-state database:

LSP ID	Sequence	Checksum	Lifetime	Attributes
R1.00-00	0x136	0x46e5	1046	L1 L2
R4.00-00	0x781	0xf65e	768	L1 L2
R3.00-00	0x7f2	0x871b	764	L1 L2
3 LSPs				

```
user@R3>show isis database level 2 igp-instance metro-b
```

IS-IS level 2 link-state database:

LSP ID	Sequence	Checksum	Lifetime	Attributes
R2.00-00	0x13a	0x7997	1013	L1 L2
R4.00-00	0x781	0x86ba	771	L1 L2
R3.00-00	0x7f2	0x1288	510	L1 L2
3 LSPs				

## On R1

```
user@R1>show isis database level 2
```

IS-IS level 2 link-state database:

LSP ID	Sequence	Checksum	Lifetime	Attributes
R1.00-00	0x136	0x46e5	851	L1 L2
R4.00-00	0x781	0xf65e	571	L1 L2
R3.00-00	0x7f2	0x871b	565	L1 L2
3 LSPs				

## Meaning

This output on R3 illustrates that R3 sees the IS-IS advertisements from R4, R5, and R6 which is standard IS-IS instance. R3 also sees the IS-IS advertisements from R1 (IS-IS metro-a), R2 (IS-IS metro-b), and R4 (both IS-IS metro-a and IS-IS metro-b). Thus, you can see that R3 is a common router that redistributes IS-IS routes among the IS-IS metro-a instance, the IS-IS metro-b instance, and the standard IS-IS instance (core network).

The output on R1 illustrates that R1 sees the IS-IS advertisements only from R3 and R4. R1 does not see any IS-IS advertisements from R2. Thus, you see that IS-IS metro-a and IS-IS metro-b are separate IS-IS flooding domains. You can use this property to build more scalable networks.

## Verify the Routing Table

### Purpose

Verify the route entries in the routing table.

### Action

From operational mode, run the `show route table inet.0 route-destination address` extensive command.

### On R3

```
user@R3>show route table inet.0 192.168.100.1 extensive

inet.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
192.168.100.1/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 192.168.100.1/32 -> {192.168.2.1}
IS-IS level 1, LSP fragment 0
IS-IS level 2, LSP fragment 0
IS-IS level 1, LSP fragment 0
IS-IS level 2, LSP fragment 0
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Router, Next hop index: 601
        Address: 0xc5b21cc
        Next-hop reference count: 2
        Next hop: 192.168.2.1 via ge-0/0/2.0, selected
        Session Id: 0x140
        State: <Active Int>
        Age: 2d 18:10:36      Metric: 63
        Validation State: unverified
        ORR Generation-ID: 0
        Task: IS-IS-metro-a
        Announcement bits (3): 0-KRT 2-IS-IS 10-IS-IS-metro-b
        AS path: I
        Thread: junos-main
```

```
user@R3>show route table inet.0 192.168.100.2 extensive

inet.0: 28 destinations, 28 routes (28 active, 0 holddown, 0 hidden)
```

```

192.168.100.2/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 192.168.100.2/32 -> {192.168.3.1}
IS-IS level 1, LSP fragment 0
IS-IS level 2, LSP fragment 0
IS-IS level 1, LSP fragment 0
IS-IS level 2, LSP fragment 0
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Router, Next hop index: 602
        Address: 0xc5b2234
        Next-hop reference count: 2
        Next hop: 192.168.3.1 via ge-0/0/1.0, selected
        Session Id: 0x141
        State: <Active Int>
        Age: 2d 18:18:48      Metric: 63
        Validation State: unverified
        ORR Generation-ID: 0
        Task: IS-IS-metro-b
        Announcement bits (3): 0-KRT 2-IS-IS 4-IS-IS-metro-a
        AS path: I
        Thread: junos-main

```

## Meaning

The output illustrates that the loopback address of R1 (192.168.100.1) is mapped to the IS-IS metro-a instance (**IS-IS-metro-a**) and the loopback address of R2 (192.168.100.2) is mapped to the IS-IS metro-b instance (**IS-IS-metro-b**) as configured in R3.

### *Verify the Routes in the IS-IS Routing Table*

## Purpose

Verify the routes in the IS-IS routing table.

## Action

From operational mode, run the `show isis route` command.

## On R3

```
user@R3>show isis route
```

```
IS-IS routing table          Current version: L1: 1885 L2: 1956
```

```
IPv4/IPv6 Routes
```

```
-----
```

Prefix	L	Version	Metric	Type	Interface	NH	Via	Backup	Score
192.168.7.0/24	2	1956	126	int	ge-0/0/4.0	IPv4	R4		
192.168.8.0/24	2	1956	126	int	ge-0/0/3.0	IPv4	R5		
192.168.100.4/32	2	1956	63	int	ge-0/0/4.0	IPv4	R4		
192.168.100.5/32	2	1956	63	int	ge-0/0/3.0	IPv4	R5		
192.168.100.6/32	2	1956	126	int	ge-0/0/3.0	IPv4	R5		
					ge-0/0/4.0	IPv4	R4		

```
user@R3>show isis route igp-instance metro-a
```

```
IS-IS routing table          Current version: L1: 1889 L2: 1961
```

```
IPv4/IPv6 Routes
```

```
-----
```

Prefix	L	Version	Metric	Type	Interface	NH	Via	Backup	Score
192.168.1.0/24	2	1961	126	int	ge-0/0/4.1	IPv4	R4		
					ge-0/0/2.0	IPv4	R1		
192.168.100.1/32	2	1961	63	int	ge-0/0/2.0	IPv4	R1		

```
user@R3>show isis route igp-instance metro-b
```

```
IS-IS routing table          Current version: L1: 1892 L2: 1949
```

```
IPv4/IPv6 Routes
```

```
-----
```

Prefix	L	Version	Metric	Type	Interface	NH	Via	Backup	Score
192.168.4.0/24	2	1949	126	int	ge-0/0/4.2	IPv4	R4		
					ge-0/0/1.0	IPv4	R2		
192.168.100.2/32	2	1949	63	int	ge-0/0/1.0	IPv4	R2		

## On R1

```
user@R1>show isis route
```

```
IS-IS routing table          Current version: L1: 313 L2: 392
```

## IPv4/IPv6 Routes

-----

Prefix	L	Version	Metric	Type	Interface	NH	Via	Backup	Score
192.168.9.0/24	2	392	126	int	ge-0/0/2.0	IPV4	R3		
					ge-0/0/1.0	IPV4	R4		
192.168.100.2/32	2	392	126	int	ge-0/0/2.0	IPV4	R3		
					ge-0/0/1.0	IPV4	R4		
192.168.100.3/32	2	392	73	int	ge-0/0/2.0	IPV4	R3		
192.168.100.4/32	2	392	73	int	ge-0/0/1.0	IPV4	R4		
192.168.100.5/32	2	392	126	int	ge-0/0/2.0	IPV4	R3		
					ge-0/0/1.0	IPV4	R4		
192.168.100.6/32	2	392	126	int	ge-0/0/2.0	IPV4	R3		
					ge-0/0/1.0	IPV4	R4		

**Meaning**

The output on R3 shows the loopback addresses and the IS-IS instance mapping information of R1, R2, R4, R5, and R6.

The output on R1 shows the loopback addresses of R2, R3, R4, R5, and R6.

**Verify IS-IS Interfaces****Purpose**

Verify the status information about IS-IS-enabled interfaces.

**Action**

From operational mode, run the `show isis interface` command.

**On R3**

```
user@R3>show isis interface
```

```
IS-IS interface database:
```

Interface	L	CirID	Level 1	DR	Level 2	DR	L1/L2	Metric
ge-0/0/3.0	2	0x1	Disabled		Point to Point		10/100	

ge-0/0/4.0	2	0x1 Disabled	Point to Point	10/100
lo0.0	3	0x1 Passive	Passive	0/0

```
user@R3>show isis interface igp-instance metro-a
```

IS-IS interface database:

Interface	L	CirID	Level 1 DR	Level 2 DR	L1/L2 Metric
ge-0/0/2.0	2	0x1	Disabled	Point to Point	10/100
ge-0/0/4.1	2	0x1	Disabled	Point to Point	10/100

```
user@R3>show isis interface igp-instance metro-b
```

IS-IS interface database:

Interface	L	CirID	Level 1 DR	Level 2 DR	L1/L2 Metric
ge-0/0/1.0	2	0x1	Disabled	Point to Point	10/100
ge-0/0/4.2	2	0x1	Disabled	Point to Point	10/100

## On R1

```
user@R1>show isis interface
```

IS-IS interface database:

Interface	L	CirID	Level 1 DR	Level 2 DR	L1/L2 Metric
ge-0/0/1.0	2	0x1	Disabled	Point to Point	10/100
ge-0/0/2.0	2	0x1	Disabled	Point to Point	10/100
lo0.0	3	0x1	Passive	Passive	0/0

## Meaning

The output shows the interfaces mapped to different IS-IS instances.



# Example: Configure Multiple Independent Instances of OSPFv2 with Segment Routing

## SUMMARY

Use this example to configure multiple IGP instances of OSPFv2 with segment routing.

## IN THIS SECTION

- [Example Prerequisites | 67](#)
- [Before You Begin | 68](#)
- [Functional Overview | 68](#)
- [Topology Overview | 68](#)
- [Topology Illustration | 69](#)
- [R2 Configuration Steps | 69](#)
- [Verification | 72](#)
- [Appendix 1: Set Commands on All Devices | 78](#)



**NOTE:** Our content testing team has validated and updated this example.



**TIP:**  
**Table 1: Readability Score and Time Estimates**

Reading Time	30 minutes
Configuration Time	20 minutes

## Example Prerequisites

Hardware requirements	Three MX Series routers.
Software requirements	Junos OS Release 24.4R1 or later running on all devices.

## Before You Begin

<b>Benefits</b>	Configuring multiple independent instances of OSPFv2 with segment routing enhances network flexibility, scalability, and control over traffic engineering, especially in large and complex networks.
<b>Know more</b>	<a href="#">Multiple Independent IGP Instances of OSPFv2</a>

## Functional Overview

<b>Technologies used</b>	<ul style="list-style-type: none"> <li>• Routing Protocols:OSPF</li> <li>• Segment Routing with Multiprotocol Label Switching (SR-MPLS)</li> <li>• VLAN Tagging</li> </ul>
<b>Primary verification tasks</b>	<ul style="list-style-type: none"> <li>• Verify that multiple independent OSPF instances are running.</li> <li>• Verify the OSPF segment routing database for different prefix-SIDs advertised by the multiple instances of OSPF</li> </ul>

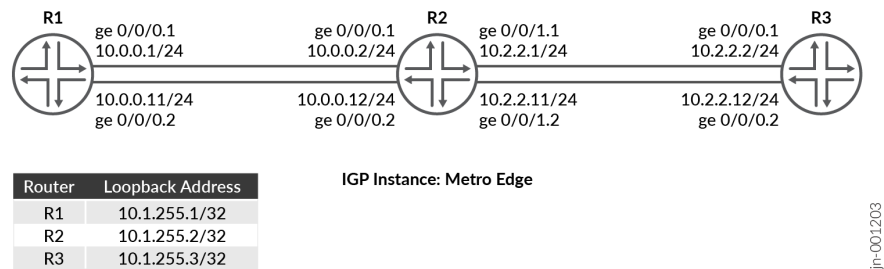
## Topology Overview

This configuration example depicts three devices R1, R2, and R3. There are two sub-interfaces configured between device R1 and device R2 and between device R2 and device R3. Each device runs multiple OSPF instances with segment routing enabled. We configure SR-MPLS to provide path control through the network. There are OSPF instances named `metro-edge` running on each of the two subinterfaces of the devices.

Hostname	Role	Function
R1, R2, and R3	The devices have multi-instance OSPF configured on the subinterfaces, with segment routing enabled.	The devices participate in OSPF multi-instances, advertise routes, and forward traffic using prefix-SIDs to other devices.

Topology Illustration

Figure 7:



R2 Configuration Steps

For complete sample configurations on R2, see: ["Appendix 1: Set Commands on All Devices" on page 78](#)

This section highlights the main configuration tasks needed to configure the R2 device for this example.

1.
  - a. Configure the basic device settings such as hostname, enhanced-ip mode, IPv4 addresses on the logical units of the device interfaces.
  - b. Configure the loopback interface with an IP address and enable MPLS.
  - c. Configure the router ID and autonomous system (AS) number to propagate routing information within a set of routing devices that belong to the same AS.
  - d. Enable VLAN tagging and configure the logical units of both the interfaces with different VLAN IDs.
  - e. Enable MPLS on each logical unit. Configure the maximum number of MPLS labels that can be applied to outgoing packets on logical units of each interface.
  - f. Define a policy to load balance packets and apply the per-packet policy to enable load balancing of traffic.
  - g. Configure a policy statement that matches routes based on the exact prefix and assign a segment identifier to the matched route.

- h. Configure MPLS traffic engineering, segment routing global block (SRGB) label range at the edit protocol mpls hierarchy level to ensure the labels are more predictable across segment routing domain, MPLS label range to assign labels from the configured srgb labels for the links.

```
[edit]
set system host-name R2
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 1 family inet address 10.0.0.2/24
set interfaces ge-0/0/0 unit 2 family inet address 10.0.0.12/24
set interfaces ge-0/0/1 unit 1 family inet address 10.2.2.1/24
set interfaces ge-0/0/1 unit 2 family inet address 10.2.2.11/24
```

```
[edit]
set interfaces lo0 unit 0 family inet address 10.1.255.2/32
set interfaces lo0 unit 0 family mpls
```

```
[edit]
set routing-options router-id 10.1.255.2
set routing-options autonomous-system 100
```

```
[edit]
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 2 vlan-id 2
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 unit 1 vlan-id 1
set interfaces ge-0/0/1 unit 2 vlan-id 2
```

```
[edit]
set interfaces ge-0/0/0 unit 1 family mpls maximum-labels 5
set interfaces ge-0/0/0 unit 2 family mpls maximum-labels 5
```

```
set interfaces ge-0/0/1 unit 1 family mpls maximum-labels 5
set interfaces ge-0/0/1 unit 2 family mpls maximum-labels 5
```

```
[edit]
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement pplb then accept
set routing-options forwarding-table export pplb
```

```
[edit]
set policy-options policy-statement prefix-sid term 1 from route-filter 10.1.255.2/32 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1001
set policy-options policy-statement prefix-sid term 1 then accept
```

```
[edit]
set protocols mpls traffic-engineering
set protocols mpls label-range srgb-label-range 800000 879999
set protocols mpls label-range static-label-range 60001 100000
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
```

2. Configure the ospf-instance metro-edge on the subinterfaces (connecting from R2 to R1 and from R2 to R3).

```
[edit]
set protocols ospf-instance metro-edge area 0.0.0.0 interface all
```

3. Enable the OSPF metro-edge instance to use segment routing with prefix-sids.

```
[edit]
set protocols ospf-instance metro-edge source-packet-routing prefix-segment prefix-sid
```

4. Configure the IPv4 index value of the node segment.

```
[edit]
set protocols ospf-instance metro-edge source-packet-routing node-segment ipv4-index 1
```

5. Configure the loopback address of the OSPF metro-edge instance as passive and disable the management interface (fxp0.0).

[edit]

```
set protocols ospf-instance metro-edge area 0.0.0.0 interface lo0.0 passive
set protocols ospf-instance metro-edge area 0.0.0.0 interface fxp0.0 disable
```

## Verification

### IN THIS SECTION

- [Verify the Routing Table | 73](#)
- [Verify OSPF Advertisements | 74](#)
- [Verify the Routes in the OSPF Routing Table | 75](#)
- [Verify the OSPF segment routing database | 76](#)
- [Verify the OSPF Interfaces | 77](#)
- [Verify the OSPF Neighbor | 77](#)

Command	Verification Task
show route protocol ospf table inet.0 extensive	<ul style="list-style-type: none"> <li>• Verify the route entries in the routing table.</li> <li>• Verify the loopback address of R1 and R3 is mapped to the igp-instance as configured in R2.</li> </ul>
show ospf spring sid- database igp-instance <i>igp- instance</i>	Verify the OSPF segment routing database for the OSPF instance.
show ospf neighbor igp- instance <i>igp-instance</i>	Verify neighbors for the specific OSPF instance.
show ospf database igp- instance <i>igp-instance</i>	Verify the OSPF advertisement entries in the OSPF link-state database (LSDB) associated with the IGP instance.
show ospf interface igp- instance <i>igp-instance</i>	Verify the interfaces mapped to the IGP instance.
show ospf route igp- instance <i>igp-instance</i>	Verify the routes and OSPF instance mapping information of R1 and R3.

## Verify the Routing Table

### Purpose

Verify the route entries in the routing table

### Action

From operational mode, run the `show route table inet.0 route-destination address extensive` command.

```
user@R2>show route protocol ospf table inet.0 10.1.255.1 extensive
inet.0: 19 destinations, 21 routes (19 active, 0 holddown, 0 hidden)
10.1.255.1/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 10.1.255.1/32 -> {list:10.0.0.1, 10.0.0.11}
    *OSPF   Preference: 10/10
           Next hop type: Router, Next hop index: 0
           Address: 0x8b32234
           Next-hop reference count: 2, Next-hop session id: 0
           Kernel Table Id: 0
           Next hop: 10.0.0.1 via ge-0/0/0.1, selected
           Session Id: 0
           Next hop: 10.0.0.11 via ge-0/0/0.2
           Session Id: 0
           State: <Active Int>
           Local AS:   100
           Age: 1w4d 16:01:19      Metric: 1
           Validation State: unverified
           Area: 0.0.0.0
           Task: OSPF-metro-edge
           Announcement bits (1): 0-KRT
           AS path: I
           Thread: junos-main
```

```
user@R2>show route protocol ospf table inet.0 10.1.255.3 extensive
inet.0: 19 destinations, 21 routes (19 active, 0 holddown, 0 hidden)
10.1.255.3/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 10.1.255.3/32 -> {list:10.2.2.2, 10.2.2.12}
    *OSPF   Preference: 10/10
           Next hop type: Router, Next hop index: 0
           Address: 0x8b316f4
```

```

Next-hop reference count: 2, Next-hop session id: 0
Kernel Table Id: 0
Next hop: 10.2.2.2 via ge-0/0/1.1, selected
Session Id: 0
Next hop: 10.2.2.12 via ge-0/0/1.2
Session Id: 0
State: <Active Int>
Local AS: 100
Age: 1w4d 16:13:55      Metric: 1
Validation State: unverified
Area: 0.0.0.0
Task: OSPF-metro-edge
Announcement bits (1): 0-KRT
AS path: I
Thread: junos-main

```

## Meaning

The output illustrates that the loopback address of R1 (10.1.255.1) and the loopback address of R3 (10.1.255.2) is mapped to the OSPF igp-instance **metro-edge** as configured in R2.

## Verify OSPF Advertisements

### Purpose

Verify the OSPF advertisement entries in the OSPF link-state database (LSDB) associated with the IGP instance.

### Action

From the operational mode, run the `show ospf database igp-instance igp-instance` command.

```

user@R2>show ospf database igp-instance metro-edge
  OSPF database, Area 0.0.0.0

```

Type	ID	Adv Rtr	Seq	Age	Opt	Cksum	Len
Router	10.1.255.1	10.1.255.1	0x80000013	1110	0x22	0xe6e9	72
Router	*10.1.255.2	10.1.255.2	0x80000015	1084	0x22	0x7be2	96
Router	10.1.255.3	10.1.255.3	0x80000013	1585	0x22	0x491	72
Network	*10.0.0.2	10.1.255.2	0x80000010	2959	0x22	0x6791	32
Network	*10.0.0.12	10.1.255.2	0x80000010	2209	0x22	0x3eb	32
Network	10.2.2.2	10.1.255.3	0x80000010	2085	0x22	0x4ba6	32



Network	10.2.2.12	10.1.255.3	0x80000010	1085	0x22	0xe601	32
OpaqArea	7.0.0.1	10.1.255.1	0x80000012	193	0x22	0x8c0	44
OpaqArea*	7.0.0.1	10.1.255.2	0x80000012	511	0x22	0x2a9b	44
OpaqArea	7.0.0.1	10.1.255.3	0x80000012	585	0x22	0x4c76	44
OpaqArea	8.0.0.1	10.1.255.1	0x80000010	2610	0x22	0x4683	48
OpaqArea*	8.0.0.1	10.1.255.2	0x80000010	2584	0x22	0xac01	52
OpaqArea	8.0.0.1	10.1.255.3	0x80000010	2584	0x22	0x7d06	52
OpaqArea	8.0.0.2	10.1.255.1	0x80000010	1860	0x22	0x4f55	48
OpaqArea*	8.0.0.2	10.1.255.2	0x80000011	334	0x22	0xf393	52
OpaqArea	8.0.0.2	10.1.255.3	0x80000011	84	0x22	0xc498	52
OpaqArea*	8.0.0.3	10.1.255.2	0x80000010	1834	0x22	0x445a	48
OpaqArea*	8.0.0.4	10.1.255.2	0x80000010	1459	0x22	0x4d2c	48

## Meaning

### *Verify the Routes in the OSPF Routing Table*

## Purpose

Verify the routes in the OSPF routing table

## Action

From the operational mode, run the `show ospf route` command.

```
user@R2>show ospf route igp-instance metro-edge
```

Topology default Route Table:

Prefix	Path	Route	NH	Metric	NextHop	NextHop
	Type	Type	Type		Interface	Address/LSP
10.1.255.1	Intra	Router	IP	1	ge-0/0/0.1	10.0.0.1
					ge-0/0/0.2	10.0.0.11
10.1.255.3	Intra	Router	IP	1	ge-0/0/1.1	10.2.2.2
					ge-0/0/1.2	10.2.2.12
10.0.0.0/24	Intra	Network	IP	1	ge-0/0/0.1	
					ge-0/0/0.2	
10.1.255.1/32	Intra	Network	IP	1	ge-0/0/0.1	10.0.0.1
					ge-0/0/0.2	10.0.0.11
10.1.255.2/32	Intra	Network	IP	0	lo0.0	
10.1.255.3/32	Intra	Network	IP	1	ge-0/0/1.1	10.2.2.2

10.2.2.0/24	Intra Network	IP	ge-0/0/1.2	10.2.2.12
			1 ge-0/0/1.1	
			ge-0/0/1.2	
299840	Intra Network	Mpls	0 ge-0/0/0.2	10.0.0.11
299840 (S=0)	Intra Network	Mpls	0 ge-0/0/0.2	10.0.0.11
299856	Intra Network	Mpls	0 ge-0/0/0.1	10.0.0.1
299856 (S=0)	Intra Network	Mpls	0 ge-0/0/0.1	10.0.0.1
299904	Intra Network	Mpls	0 ge-0/0/1.2	10.2.2.12
299904 (S=0)	Intra Network	Mpls	0 ge-0/0/1.2	10.2.2.12
299920	Intra Network	Mpls	0 ge-0/0/1.1	10.2.2.2
299920 (S=0)	Intra Network	Mpls	0 ge-0/0/1.1	10.2.2.2

## Meaning

The output on R2 shows the loopback addresses and OSPF instance mapping information of R1 and R3.

### *Verify the OSPF segment routing database*

## Purpose

Verify the OSPF segment routing database for the OSPF instance metro-edge.

## Action

From the operational mode, run the `show ospf spring sid-database igp-instance igp-instance` command.

```
user@R2>show ospf spring sid-database igp-instance metro-edge
OSPF database, Area 0.0.0.0
```

SID	Prefix	Advertised-by	Route-type
<b>1000</b>	10.1.255.1/32	10.1.255.1	Intra-Area
<b>1001</b>	10.1.255.2/32	10.1.255.2	Intra-Area
<b>1002</b>	10.1.255.3/32	10.1.255.3	Intra-Area

## Meaning

The output illustrates the multiple instances of OSPF (metro-edge) advertise prefix-SIDs.

## Verify the OSPF Interfaces

### Purpose

Verify the status information about OSPF-instance enabled interfaces.

### Action

From the operational mode, run the `show ospf interface igp-instance igp-instance` command.

```
user@R2>show ospf interface igp-instance metro-edge
```

Interface	State	Area	DR ID	BDR ID	Nbrs
ge-0/0/0.1	DR	0.0.0.0	10.1.255.2	10.1.255.1	1
ge-0/0/0.2	DR	0.0.0.0	10.1.255.2	10.1.255.1	1
ge-0/0/1.1	DR	0.0.0.0	10.1.255.2	10.1.255.3	1
ge-0/0/1.2	DR	0.0.0.0	10.1.255.2	10.1.255.3	1
lo0.0	DRother	0.0.0.0	0.0.0.0	0.0.0.0	0
lo0.0	DRother	0.0.0.0	0.0.0.0	0.0.0.0	0

### Meaning

The output shows the subinterfaces of R2 mapped to the OSPF instances (metro-edge).

## Verify the OSPF Neighbor

### Purpose

Verify the adjacencies between the configured links.

### Action

From the operational mode, run the `show ospf neighbor igp-instance igp-instance` command.

```
user@R2>show ospf neighbor igp-instance metro-edge
```

Address	Interface	State	ID	Pri	Dead
10.0.0.1	ge-0/0/0.1	Full	10.1.255.1	128	35
10.0.0.11	ge-0/0/0.2	Full	10.1.255.1	128	39
10.2.2.2	ge-0/0/1.1	Full	10.1.255.3	128	33
10.2.2.12	ge-0/0/1.2	Full	10.1.255.3	128	36

## Meaning

Device R2 has established adjacency with Device R1 and Device R3 and as indicated by the State output field which is Full.

## Appendix 1: Set Commands on All Devices

### IN THIS SECTION

- R1 | 78
- R2 | 79
- R3 | 80

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

### ***R1***

```
set system host-name R1
set interfaces ge-0/0/0 unit 1 family inet address 10.0.0.1/24
set interfaces ge-0/0/0 unit 2 family inet address 10.0.0.11/24
set interfaces ge-0/0/0 unit 2 enable
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 2 vlan-id 2
set interfaces ge-0/0/0 unit 1 family mpls maximum-labels 5
set interfaces ge-0/0/0 unit 2 family mpls maximum-labels 5
set interfaces lo0 unit 0 family inet address 10.1.255.1/32
set interfaces lo0 unit 0 family mpls
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement pplb then accept
set policy-options policy-statement prefix-sid term 1 from route-filter 10.1.255.1/32 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1000
set policy-options policy-statement prefix-sid term 1 then accept
set routing-options router-id 10.1.255.1
set routing-options autonomous-system 100
set routing-options forwarding-table export pplb
```

```

set protocols mpls traffic-engineering
set protocols mpls label-range srgb-label-range 800000 879999
set protocols mpls label-range static-label-range 60001 100000
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ospf-instance metro-edge source-packet-routing prefix-segment prefix-sid
set protocols ospf-instance metro-edge source-packet-routing node-segment ipv4-index 0
set protocols ospf-instance metro-edge area 0.0.0.0 interface all
set protocols ospf-instance metro-edge area 0.0.0.0 interface lo0.0 passive
set protocols ospf-instance metro-edge area 0.0.0.0 interface fxp0.0 disable

```

## R2

```

set system host-name R2
set interfaces ge-0/0/0 unit 1 family inet address 10.0.0.2/24
set interfaces ge-0/0/0 unit 2 family inet address 10.0.0.12/24
set interfaces ge-0/0/1 unit 1 family inet address 10.2.2.1/24
set interfaces ge-0/0/1 unit 2 family inet address 10.2.2.11/24
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 2 vlan-id 2
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 unit 1 vlan-id 1
set interfaces ge-0/0/1 unit 2 vlan-id 2
set interfaces ge-0/0/0 unit 1 family mpls maximum-labels 5
set interfaces ge-0/0/0 unit 2 family mpls maximum-labels 5
set interfaces ge-0/0/1 unit 1 family mpls maximum-labels 5
set interfaces ge-0/0/1 unit 2 family mpls maximum-labels 5
set interfaces lo0 unit 0 family inet address 10.1.255.2/32
set interfaces lo0 unit 0 family mpls
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement pplb then accept
set policy-options policy-statement prefix-sid term 1 from route-filter 10.1.255.2/32 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1001
set policy-options policy-statement prefix-sid term 1 then accept
set routing-options router-id 10.1.255.2
set routing-options autonomous-system 100
set routing-options forwarding-table export pplb
set protocols mpls traffic-engineering
set protocols mpls label-range srgb-label-range 800000 879999
set protocols mpls label-range static-label-range 60001 100000

```

```

set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ospf-instance metro-edge source-packet-routing prefix-segment prefix-sid
set protocols ospf-instance metro-edge source-packet-routing node-segment ipv4-index 1
set protocols ospf-instance metro-edge area 0.0.0.0 interface all
set protocols ospf-instance metro-edge area 0.0.0.0 interface lo0.0 passive
set protocols ospf-instance metro-edge area 0.0.0.0 interface fxp0.0 disable

```

### R3

```

set system host-name R3
set interfaces ge-0/0/0 unit 1 family inet address 10.2.2.2/24
set interfaces ge-0/0/0 unit 2 family inet address 10.2.2.12/24
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 2 vlan-id 2
set interfaces ge-0/0/0 unit 1 family mpls maximum-labels 5
set interfaces ge-0/0/0 unit 2 family mpls maximum-labels 5
set interfaces lo0 unit 0 family inet address 10.1.255.3/32
set interfaces lo0 unit 0 family mpls
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement pplb then accept
set policy-options policy-statement prefix-sid term 1 from route-filter 10.1.255.3/32 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1002
set policy-options policy-statement prefix-sid term 1 then accept
set routing-options router-id 10.1.255.3
set routing-options autonomous-system 100
set routing-options forwarding-table export pplb
set protocols mpls traffic-engineering
set protocols mpls label-range srgb-label-range 800000 879999
set protocols mpls label-range static-label-range 60001 100000
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ospf-instance metro-edge source-packet-routing prefix-segment prefix-sid
set protocols ospf-instance metro-edge source-packet-routing node-segment ipv4-index 2
set protocols ospf-instance metro-edge area 0.0.0.0 interface lo0.0 passive
set protocols ospf-instance metro-edge area 0.0.0.0 interface all
set protocols ospf-instance metro-edge area 0.0.0.0 interface fxp0.0 disable

```

### Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
change-completed	Starting in Junos OS and Junos OS Evolved Release 24.2R1, you can configure and run multiple independent IGP instances of OSPFv2 simultaneously on a router.
change-completed	Starting in Junos OS and Junos OS Evolved Release 24.4R1, you can configure and run multiple independent interior gateway protocol (IGP) instances of OSPFv2 with segment routing (SR) on a router.

# Configuring IS-IS Authentication and Checksums

## IN THIS CHAPTER

- [Configuring IS-IS Authentication | 82](#)
- [Configuring IS-IS Authentication Without Network-Wide Deployment | 84](#)
- [Understanding Hitless Authentication Key Rollover for IS-IS | 84](#)
- [Example: Configuring Hitless Authentication Key Rollover for IS-IS | 86](#)
- [Understanding Checksums on IS-IS Interfaces for Error Checking | 94](#)
- [Example: Enabling Packet Checksums on IS-IS Interfaces for Error Checking | 94](#)

## Configuring IS-IS Authentication

All IS-IS protocol exchanges can be authenticated to guarantee that only trusted routing devices participate in the autonomous system (AS) routing. By default, IS-IS authentication is disabled on the routing device.

To configure IS-IS authentication, you must define an authentication password and specify the authentication type.

You can configure one of the following authentication methods:

- Simple authentication—Uses a text password that is included in the transmitted packet. The receiving routing device uses an authentication key (password) to verify the packet. Simple authentication is included for compatibility with existing IS-IS implementations. However, we recommend that you do *not* use this authentication method because it is insecure (the text can be “sniffed”).



**CAUTION:** A simple password that exceeds 254 characters is truncated.

- HMAC-MD5 authentication—Uses an iterated cryptographic hash function. The receiving routing device uses an authentication key (password) to verify the packet.

You can also configure more fine-grained interface-level authentication for hello packets.



To enable authentication and specify an authentication method, include the `authentication-type` statement, specifying the `simple` or `md5` authentication type:

```
authentication-type authentication;
```

You can configure this statement at the `[edit protocols isis]` or at the `[edit protocols isis interface interface-name]` hierarchy levels.

To configure a password, include the `authentication-key` statement. The authentication password for all routing devices in a domain must be the same.

```
authentication-key key;
```

To configure hitless authentication key rollover, include the `authentication-key-chain` (Protocols IS-IS) statement.

The password can contain up to 255 characters. If you include spaces, enclose all characters in quotation marks (" ").

If you are using the Junos OS IS-IS software with another implementation of IS-IS, the other implementation must be configured to use the same password for the domain, the area, and all interfaces that are shared with a Junos OS implementation.

Authentication of hello packets, partial sequence number PDU (PSNP), and complete sequence number PDU (CSNP) can be suppressed to enable interoperability with the routing software of different vendors. Different vendors handle authentication in various ways, and suppressing authentication for different PDU types might be the simplest way to allow compatibility within the same network.

To configure IS-IS to generate authenticated packets, but not to check the authentication on received packets, include the `no-authentication-check` statement:

```
no-authentication-check;
```

To suppress authentication of IS-IS hello packets, include the `no-hello-authentication` statement:

```
no-hello-authentication;
```

To suppress authentication of PSNPs, include the `no-psnp-authentication` statement:

```
no-psnp-authentication;
```

To suppress authentication of CSNPs, include the `no-csnp-authentication` statement:

```
no-csnp-authentication;
```



**NOTE:** The authentication and the no-authentication statements must be configured at the same hierarchy level. Configuring authentication at the [edit protocols isis interface *interface-name*] hierarchy level and configuring no-authentication at the [edit protocols isis] hierarchy level has no effect.

## RELATED DOCUMENTATION

[Configuring IS-IS Authentication Without Network-Wide Deployment | 84](#)

## Configuring IS-IS Authentication Without Network-Wide Deployment

To allow the use of authentication without requiring network-wide deployment, include the `loose-authentication-check` statement:

```
loose-authentication-check;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

## RELATED DOCUMENTATION

[Understanding Hitless Authentication Key Rollover for IS-IS | 84](#)

[Example: Configuring Hitless Authentication Key Rollover for IS-IS | 86](#)

## Understanding Hitless Authentication Key Rollover for IS-IS

IS-IS protocol exchanges can be authenticated to guarantee that only trusted routing devices participate in routing. By default, authentication is disabled. The authentication algorithm creates an encoded

checksum that is included in the transmitted packet. The receiving routing device uses an authentication key (password) to verify the packet's checksum.

If you configure authentication for all peers, each peer in that group inherits the group's authentication.

You can update authentication keys without resetting any IS-IS neighbor sessions. This is referred to as *hitless authentication key rollover*.

Hitless authentication key rollover uses authentication keychains, which consist of the authentication keys that are being updated. The keychain includes multiple keys. Each key in the keychain has a unique start time. At the next key's start time, a rollover occurs from the current key to the next key, and the next key becomes the current key.

You can choose the algorithm through which authentication is established. You can configure MD5 or SHA-1 authentication. Starting in Junos OS Release 24.2R1, we extend support to IS-IS keychain with the following hash functions:

- HMAC-SHA2-224
- HMAC-SHA2-256
- HMAC-SHA2-384
- HMAC-SHA2-512

You associate a keychain and the authentication algorithm with an IS-IS neighboring session. Each key contains an identifier and a secret password.

The sending peer chooses the active key based on the system time and the start times of the keys in the keychain. The receiving peer determines the key with which it authenticates based on the incoming key identifier.

You can configure either RFC 5304-based encoding or RFC 5310-based encoding for the IS-IS protocol transmission encoding format.

## RELATED DOCUMENTATION

| [Example: Configuring Hitless Authentication Key Rollover for IS-IS](#) | 86

## Example: Configuring Hitless Authentication Key Rollover for IS-IS

### IN THIS SECTION

- [Requirements | 86](#)
- [Overview | 86](#)
- [Configuration | 88](#)
- [Verification | 93](#)

This example shows how to configure hitless authentication key rollover for IS-IS.

### Requirements

No special configuration beyond device initialization is required before configuring hitless authentication key rollover for IS-IS.

### Overview

#### IN THIS SECTION

- [Topology | 87](#)

Authentication guarantees that only trusted routers participate in routing updates. This keychain authentication method is referred to as hitless because the keys roll over from one to the next without resetting any peering sessions or interrupting the routing protocol. Junos OS supports both RFC 5304, *IS-IS Cryptographic Authentication* and RFC 5310, *IS-IS Generic Cryptographic Authentication*.

This example includes the following statements for configuring the keychain:

- **algorithm**—For each key in the keychain, you can specify an encryption algorithm. The algorithm can be SHA-1 or MD-5.
- **key**—A keychain can have multiple keys. Each key within a keychain must be identified by a unique integer value. The range of valid identifier values is from 0 through 63.
- **key-chain**—For each keychain, you must specify a name. This example defines two keychains: **base-key-global** and **base-key-inter**.

- **options**—For each key in the keychain, you can specify the encoding for the message authentication code: **isis-enhanced** or **basic**. The basic (RFC 5304) operation is enabled by default.

When you configure the **isis-enhanced** option, Junos OS sends RFC 5310-encoded routing protocol packets and accepts both RFC 5304-encoded and RFC 5310-encoded routing protocol packets that are received from other devices.

When you configure **basic** (or do not include the **options** statement in the key configuration) Junos OS sends and receives RFC 5304-encoded routing protocols packets, and drops 5310-encoded routing protocol packets that are received from other devices.

Because this setting is for IS-IS only, the TCP and the BFD protocols ignore the encoding option configured in the key.

- **secret**—For each key in the keychain, you must set a secret password. This password can be entered in either encrypted or plain text format in the **secret** statement. It is always displayed in encrypted format.
- **start-time**—Each key must specify a start time based on UTC using the ISO 8601 format. Control gets passed from one key to the next. When a configured start time arrives (based on the routing device's clock), the key with that start time becomes active. Start times are specified in the local time zone for a routing device and must be unique within the key chain.

You can apply a keychain globally to all interfaces or more granularly to specific interfaces.

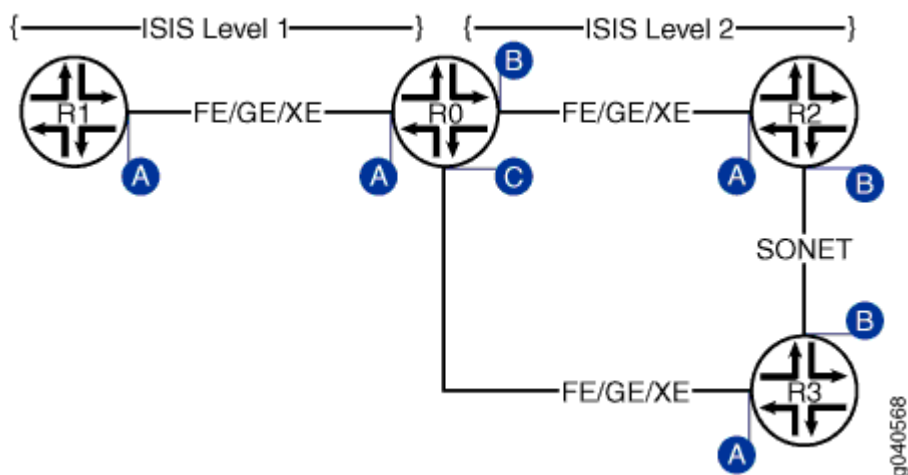
This example includes the following statements for applying the keychain to all interfaces or to particular interfaces:

- **authentication-key-chain**—Enables you to apply a keychain at the global IS-IS level for all Level 1 or all Level 2 interfaces.
- **hello-authentication-key-chain**—Enables you to apply a keychain at the individual IS-IS interface level. The interface configuration overrides the global configuration.

## Topology

[Figure 8 on page 88](#) shows the topology used in the example.

Figure 8: Hitless Authentication Key Rollover for IS-IS



This example shows the configuration for Router R0.

## Configuration

### IN THIS SECTION

● Procedure | 88

● Results | 91

## Procedure

### CLI Quick Configuration for R0

To quickly configure the hitless authentication key rollover for IS-IS, copy the following commands and paste the commands into the CLI.

```
[edit]
set interfaces ge-0/0/0 unit 0 description "interface A"
set interfaces ge-0/0/0 unit 0 family inet address 10.0.0.1/30
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family inet6 address fe80::200:f8ff:fe21:67cf/128
set interfaces ge-0/0/1 unit 0 description "interface B"
set interfaces ge-0/0/1 unit 0 family inet address 10.0.0.5/30
set interfaces ge-0/0/1 unit 0 family iso
```

```

set interfaces ge-0/0/1 unit 0 family inet6 address 10FB::C:ABC:1F0C:44DA/128
set interfaces ge-0/0/2 unit 0 description "interface C"
set interfaces ge-0/0/2 unit 0 family inet address 10.0.0.9/30
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/2 unit 0 family inet6 address ff06::c3/128
set security authentication-key-chains key-chain base-key-global key 63 secret "$ABC123"
set security authentication-key-chains key-chain base-key-global key 63 start-time
"2011-8-6.06:54:00-0700"
set security authentication-key-chains key-chain base-key-global key 63 algorithm hmac-sha-1
set security authentication-key-chains key-chain base-key-global key 63 options isis-enhanced
set security authentication-key-chains key-chain base-key-global key 64 secret "$ABC1234"
set security authentication-key-chains key-chain base-key-global key 64 start-time
"2011-10-6.06:54:00-0700"
set security authentication-key-chains key-chain base-key-global key 64 algorithm hmac-sha-1
set security authentication-key-chains key-chain base-key-global key 64 options isis-enhanced
set security authentication-key-chains key-chain base-key-inter key 0 secret "$ABC123"
set security authentication-key-chains key-chain base-key-inter key 0 start-time
"2011-8-6.06:54:00-0700"
set security authentication-key-chains key-chain base-key-inter key 0 algorithm md5
set security authentication-key-chains key-chain base-key-inter key 0 options basic
set security authentication-key-chains key-chain base-key-inter key 1 secret "$ABC1234"
set security authentication-key-chains key-chain base-key-inter key 1 start-time
"2011-10-6.06:54:00-0700"
set security authentication-key-chains key-chain base-key-inter key 1 algorithm md5
set security authentication-key-chains key-chain base-key-inter key 1 options basic
set protocols isis level 2 authentication-key-chain base-key-global
set protocols isis interface ge-0/0/0.0 level 1 hello-authentication-key-chain base-key-inter

```

## Step-by-Step Procedure

To configure hitless authentication key rollover for IS-IS:

1. Configure the Router R0 interfaces.

```

[edit]
user@host# edit interfaces ge-0/0/0 unit 0
[edit interfaces ge-0/0/0 unit 0]
user@host# set description "interface A"
user@host# set family inet address 10.0.0.1/30
user@host# set family iso
user@host# set family inet6 address fe80::200:f8ff:fe21:67cf/128
user@host# exit

```

```

[edit]
user@host# edit interfaces ge-0/0/1 unit 0
[edit interfaces ge-0/0/1 unit 0]
user@host# set interfaces ge-0/0/1 unit 0 description "interface B"
user@host# set interfaces ge-0/0/1 unit 0 family inet address 10.0.0.5/30
user@host# set interfaces ge-0/0/1 unit 0 family iso
user@host# set interfaces ge-0/0/1 unit 0 family inet6 address 10FB::C:ABC:1F0C:44DA/128
user@host# exit
[edit]
user@host# edit interfaces ge-0/0/2 unit 0
[edit interfaces ge-0/0/2 unit 0]
user@host# set description "interface C"
user@host# set family inet address 10.0.0.9/30
user@host# set interfaces ge-0/0/2 unit 0 family iso
user@host# set interfaces ge-0/0/2 unit 0 family inet6 address ff06::c3/128
user@host# exit

```

2. Configure one or more authentication key chains and keys. In this example we demonstrate the use of both a global and interface level key chain, both having two keys. The global key chain is applied to all ISIS Level 2 interfaces. This key chain authenticates both hellos and LSP exchanges. The interface key chain is applied specifically to the ge-0/0/0 interface (Interface A) for ISIS Level 1 and is used only for authenticating hello exchanges.

```

[edit]
user@host# edit security authentication-key-chains key-chain base-key-global
[edit security authentication-key-chains key-chain base-key-global]
user@host# set key 63 secret "$ABC123"
user@host# set key 63 start-time "2011-8-6.06:54:00-0700"
user@host# set key 63 algorithm hmac-sha-1
user@host# set key 63 options isis-enhanced

user@host# set key 64 secret "$ABC1234"
user@host# set key 64 start-time "2011-10-6.06:54:00-0700"
user@host# set key 64 algorithm hmac-sha-1
user@host# set key 64 options isis-enhanced

user@host# exit
[edit]
user@host# edit security authentication-key-chains key-chain base-key-inter
[edit security authentication-key-chains key-chain base-key-inter]
user@host# set key 0 secret "$ABC123"
user@host# set key 0 start-time "2011-8-6.06:54:00-0700"

```



```

user@host# set key 0 algorithm md5
user@host# set key 0 options basic

user@host# set key 1 secret "$ABC1234"
user@host# set key 1 start-time "2011-10-6.06:54:00-0700"
user@host# set key 1 algorithm md5
user@host# set key 1 options basic
user@host# exit

```

3. Apply the base-key-global key chain to all Level 2 ISIS interfaces on Router R0.

```

[edit]
user@host# edit protocols isis level 2
[edit protocols isis level 1]
set authentication-key-chain base-key-global
user@host# exit

```

4. Apply the base-key-inter key chain for ISIS hello authentication at Level 1 to the **ge-0/0/0.0** interface on Router R0.

```

[edit]
user@host# edit protocols isis interface ge-0/0/0.0 level 1
[edit protocols isis interface ge-0/0/0.0 level 1]
set hello-authentication-key-chain base-key-inter
user@host# exit

```

5. If you are done configuring the device, commit the configuration.

```

[edit]
user@host# commit

```

## Results

Confirm your configuration by entering the **show interfaces**, **show protocols**, and **show security** commands.

```

user@host# show interfaces
ge-0/0/0 {

```

```

    unit 0 {
        description "interface A";
        family inet {
            address 10.0.0.1/30;
        }
        family iso;
        family inet6 {
            address fe80::200:f8ff:fe21:67cf/128;
        }
    }
}
ge-0/0/1 {
    unit 0 {
        description "interface B";
        family inet {
            address 10.0.0.5/30;
        }
        family iso;
        family inet6 {
            address 10FB::C:ABC:1F0C:44DA/128;
        }
    }
}
ge-0/0/2 {
    unit 0 {
        description "interface C";
        family inet {
            address 10.0.0.9/30;
        }
        family iso;
        family inet6 {
            address ff06::c3/128;
        }
    }
}
}

```

```
user@host# show protocols
```

```

isis {
    level 2 authentication-key-chain base-key-global;
    interface ge-0/0/0.0 {
        level 1 hello-authentication-key-chain base-key-inter;
    }
}

```

```

    }
}

```

```

user@host# show security
authentication-key-chains {
  key-chain base-key-global {
    key 63 {
      secret "ABC123";
      start-time "2011-8-6.06:54:00-0700";
      algorithm hmac-sha-1;
      options isis-enhanced;
    }

    key 64 {
      secret "ABC1234";
      start-time "2011-10-6.06:54:00-0700";
      algorithm hmac-sha-1;
      options isis-enhanced;
    }
  }
  key-chain base-key-inter {
    key 0 {
      secret "$ABC123";
      start-time "2011-8-6.06:54:00-0700";
      algorithm md5;
      options basic;
    }
    key 1 {
      secret "$ABC1234";
      start-time "2011-10-6.06:54:00-0700";
      algorithm md5;
      options basic;
    }
  }
}
}

```

## Verification

To verify the configuration, run the following commands:

- show isis authentication
- show security keychain

## RELATED DOCUMENTATION

[Understanding Hitless Authentication Key Rollover for IS-IS | 84](#)

## Understanding Checksums on IS-IS Interfaces for Error Checking

The checksum enables IS-IS to check at the receiver if the IS-IS protocol frames have become corrupted while traversing the network.

Sometimes corrupt IS-IS protocol frames can go undetected. If routing control traffic becomes corrupted, it is likely that user payload traffic might be corrupted, too. This can lead to unacceptable throughput. To prevent corrupt frames from going undetected, we recommend enabling checksumming on the IS-IS interfaces.

To review, IS-IS hello (IIH) PDUs establish adjacencies with other routing devices. A partial sequence number PDU (PSNP) is used by an IS-IS router to request link-state PDU information from a neighboring router. The complete sequence number PDU (CSNP) lists all the link-state PDUs in the link-state database.

The original specification for IS-IS does not provide checksums for IIHs, CSNPs, and PSNPs.

RFC 3358, *Optional Checksums in Intermediate System to Intermediate System (IS-IS)* introduced an optional type, length, and value (TLV) tuple that provides checksums for IIHs, PSNPs, and CSNPs.

Junos OS supports the checksumming TLV on a per-interface basis.

## RELATED DOCUMENTATION

[Example: Enabling Packet Checksums on IS-IS Interfaces for Error Checking | 94](#)

## Example: Enabling Packet Checksums on IS-IS Interfaces for Error Checking

### IN THIS SECTION

- [Requirements | 95](#)
- [Overview | 95](#)

- Configuration | 96
- Verification | 97

This example shows how to enable packet checksums for IS-IS interfaces.

## Requirements

Before you begin, configure IS-IS on both routers. See ["Example: Configuring IS-IS" on page 15](#) for information about the sample IS-IS configuration.

## Overview

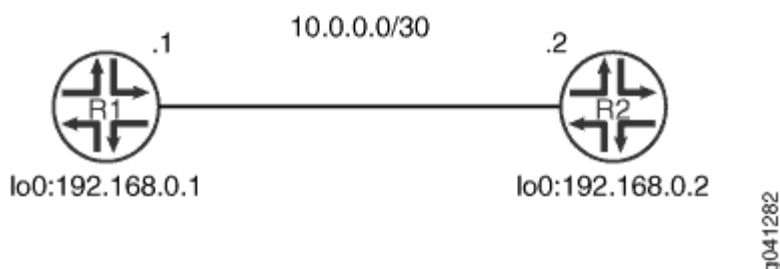
Junos OS supports IS-IS checksums as documented in RFC 3358, *Optional Checksums in Intermediate System to Intermediate System (ISIS)*.

IS-IS protocol data units (PDUs) include link-state PDUs, complete sequence number PDUs (CSNPs), partial sequence number PDUs (PSNPs), and IS-IS hello (IIH) packets. These PDUs can be corrupt due to faulty implementations of Layer 2 hardware or lack of checksums on a specific network technology. Corruption of length or type, length, and value (TLV) fields can lead to the generation of extensive numbers of empty link-state PDUs in the receiving node. Because authentication is not a replacement for a checksum mechanism, you might want to enable the optional checksum TLV on your IS-IS interfaces.

The checksum cannot be enabled with MD5 hello authentication on the same interface.

[Figure 9 on page 95](#) shows the topology used in this example.

**Figure 9: IS-IS Checksum Topology**



This example describes the steps on Device R1.

## Configuration

### IN THIS SECTION

- [Procedure](#) | 96

### Procedure

#### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R1

```
set protocols isis traceoptions file isis
set protocols isis traceoptions flag all
set protocols isis interface fe-1/2/0.1 checksum
```

#### Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure IS-IS checksums:

1. Enable checksums.

```
[edit protocols isis interface fe-1/2/0.1]
user@R1# set checksum
```

## 2. (Optional) Enable tracing for tracking checksum operations.

```
[edit protocols isis traceoptions]  
user@R1# set file isis  
user@R1# set flag all
```

## Results

From configuration mode, confirm your configuration by entering the `show protocols` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show protocols  
isis {  
  traceoptions {  
    file isis;  
    flag all;  
  }  
  interface fe-1/2/0.1 {  
    checksum;  
  }  
}
```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Verifying Checksums | 98](#)

Confirm that the configuration is working properly.

## Verifying Checksums

### Purpose

Verify that checksums are performed.

### Action

From operational mode, enter the `show log isis | match checksum` command.

```
user@R1> show log isis | match checksum

May 31 16:47:39.513267      sequence 0x49 checksum 0x8e64
May 31 16:47:39.513394      sequence 0x4e checksum 0x34b3
May 31 16:47:39.513517      sequence 0x50 checksum 0x9dcb
May 31 16:47:46.563781      sequence 0x45 checksum 0x7e1a
May 31 16:47:46.563970      sequence 0x46 checksum 0x226d
May 31 16:47:46.564104      sequence 0x52 checksum 0x99cd
May 31 16:47:46.581087      sequence 0x49 checksum 0x8e64
May 31 16:47:46.581222      sequence 0x4e checksum 0x34b3
May 31 16:47:46.581353      sequence 0x50 checksum 0x9dcb
May 31 16:47:55.799090      sequence 0x45 checksum 0x7e1a
May 31 16:47:55.799223      sequence 0x46 checksum 0x226d
May 31 16:47:55.799347      sequence 0x52 checksum 0x99cd
May 31 16:47:55.818255      sequence 0x49 checksum 0x8e64
May 31 16:47:55.818473      sequence 0x4e checksum 0x34b3
May 31 16:47:55.818606      sequence 0x50 checksum 0x9dcb
May 31 16:48:03.455816      sequence 0x49 checksum 0x8e64
May 31 16:48:03.455973      sequence 0x4e checksum 0x34b3
```

### Meaning

The output shows that checksum information is captured in the IS-IS trace log file.

## RELATED DOCUMENTATION

[Understanding Checksums on IS-IS Interfaces for Error Checking](#) | 94



## CHAPTER 4

# Configuring IS-IS Routing Policy and Route Redistribution

## IN THIS CHAPTER

- Understanding Routing Policies | 99
- Understanding Backup Selection Policy for IS-IS Protocol | 103
- Example: Configuring Backup Selection Policy for IS-IS Protocol | 105
- Configuring Backup Selection Policy for the IS-IS Protocol | 144
- Example: Redistributing OSPF Routes into IS-IS | 151
- Example: Configuring IS-IS Route Leaking from a Level 2 Area to a Level 1 Area | 165
- Handling of the IS-IS Binding SID S Flag and RFC 7794 Prefix Attribute Flags | 174
- Understanding BGP Communities, Extended Communities, and Large Communities as Routing Policy Match Conditions | 176
- Example: Configuring a Routing Policy to Redistribute BGP Routes with a Specific Community Tag into IS-IS | 178
- IS-IS Extensions to Support Route Tagging | 191
- Example: Configuring a Routing Policy to Prioritize IS-IS Routes | 193
- Configuring Overloading of Stub Networks | 205

## Understanding Routing Policies

### IN THIS SECTION

- Importing and Exporting Routes | 100
- Active and Inactive Routes | 102
- Explicitly Configured Routes | 102
- Dynamic Database | 102

For some routing platform vendors, the flow of routes occurs between various protocols. If, for example, you want to configure redistribution from RIP to OSPF, the RIP process tells the OSPF process that it has routes that might be included for redistribution. In Junos OS, there is not much direct interaction between the routing protocols. Instead, there are central gathering points where all protocols install their routing information. These are the main unicast routing tables `inet.0` and `inet6.0`.

From these tables, the routing protocols calculate the best route to each destination and place these routes in a forwarding table. These routes are then used to forward routing protocol traffic toward a destination, and they can be advertised to neighbors.

## Importing and Exporting Routes

Two terms—*import* and *export*—explain how routes move between the routing protocols and the routing table.

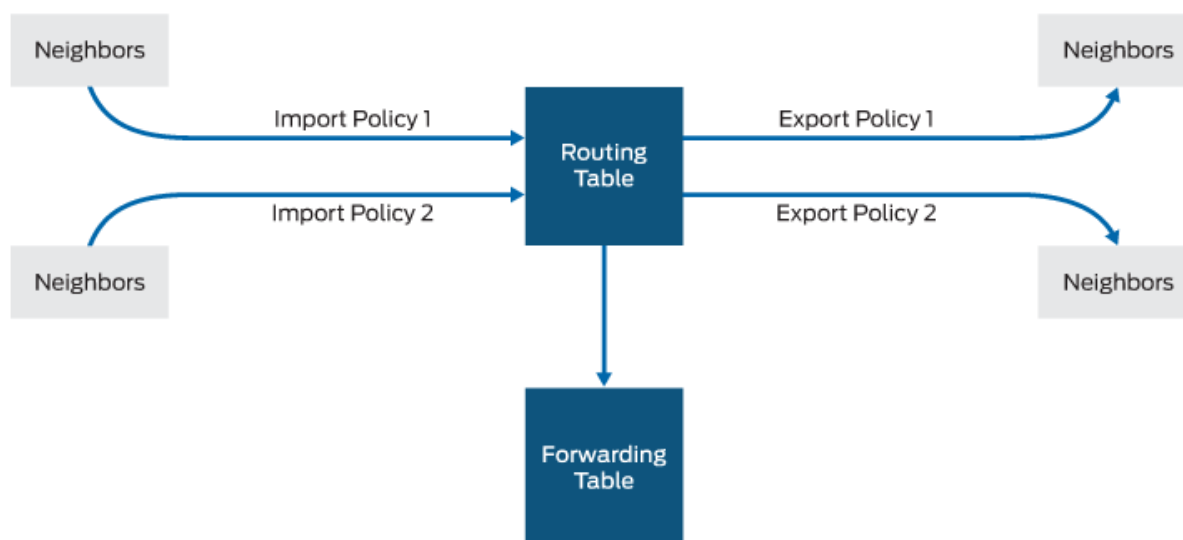
- When the Routing Engine places the routes of a routing protocol into the routing table, it is *importing* routes into the routing table.
- When the Routing Engine uses active routes from the routing table to send a protocol advertisement, it is *exporting* routes from the routing table.



**NOTE:** The process of moving routes between a routing protocol and the routing table is described always *from the point of view of the routing table*. That is, routes are *imported into* a routing table from a routing protocol and they are *exported from* a routing table to a routing protocol. Remember this distinction when working with routing policies.

As shown in [Figure 10 on page 101](#), you use import routing policies to control which routes are placed in the routing table, and export routing policies to control which routes are advertised from the routing table to neighbors.

Figure 10: Importing and Exporting Routes



In general, the routing protocols place all their routes in the routing table and advertise a limited set of routes from the routing table. The general rules for handling the routing information between the routing protocols and the routing table are known as the *routing policy framework*.

The routing policy framework is composed of default rules for each routing protocol that determine which routes the protocol places in the routing table and advertises from the routing table. The default rules for each routing protocol are known as *default routing policies*.

You can create routing policies to preempt the default policies, which are always present. A *routing policy* allows you to modify the routing policy framework to suit your needs. You can create and implement your own routing policies to do the following:

- Control which routes a routing protocol places in the routing table.
- Control which active routes a routing protocol advertises from the routing table. An *active route* is a route that is chosen from all routes in the routing table to reach a destination.
- Manipulate the route characteristics as a routing protocol places the route in the routing table or advertises the route from the routing table.

You can manipulate the route characteristics to control which route is selected as the active route to reach a destination. The active route is placed in the forwarding table and is used to forward traffic toward the route's destination. In general, the active route is also advertised to a router's neighbors.

## Active and Inactive Routes

When multiple routes for a destination exist in the routing table, the protocol selects an active route and that route is placed in the appropriate routing table. For equal-cost routes, the Junos OS places multiple next hops in the appropriate routing table.

When a protocol is exporting routes from the routing table, it exports active routes only. This applies to actions specified by both default and user-defined export policies.

When evaluating routes for export, the Routing Engine uses only active routes from the routing table. For example, if a routing table contains multiple routes to the same destination and one route has a preferable metric, only that route is evaluated. In other words, an export policy does not evaluate all routes; it evaluates only those routes that a routing protocol is allowed to advertise to a neighbor.



**NOTE:** By default, BGP advertises active routes. However, you can configure BGP to advertise *inactive routes*, which go to the same destination as other routes but have less preferable metrics.

## Explicitly Configured Routes

An *explicitly configured route* is a route that you have configured. *Direct routes* are not explicitly configured. They are created as a result of IP addresses being configured on an interface. Explicitly configured routes include aggregate, generated, local, and static routes. (An *aggregate route* is a route that distills groups of routes with common addresses into one route. A *generated route* is a route used when the routing table has no information about how to reach a particular destination. A *local route* is an IP address assigned to a router interface. A *static route* is an unchanging route to a destination.)

The policy framework software treats direct and explicitly configured routes as if they are learned through routing protocols; therefore, they can be imported into the routing table. Routes cannot be exported from the routing table to the pseudoprotocol, because this protocol is not a real routing protocol. However, aggregate, direct, generated, and static routes can be exported from the routing table to routing protocols, whereas local routes cannot.

## Dynamic Database

In Junos OS Release 9.5 and later, you can configure routing policies and certain routing policy objects in a dynamic database that is not subject to the same verification required by the standard configuration database. As a result, you can quickly commit these routing policies and policy objects, which can be referenced and applied in the standard configuration as needed. BGP is the only protocol to which you can apply routing policies that reference policies configured in the dynamic database. After a routing policy based on the dynamic database is configured and committed in the standard configuration, you can quickly make changes to existing routing policies by modifying policy objects in the dynamic

database. Because Junos OS does not validate configuration changes to the dynamic database, when you use this feature, you should test and verify all configuration changes before committing them.

## RELATED DOCUMENTATION

[Example: Configuring Dynamic Routing Policies](#)

## Understanding Backup Selection Policy for IS-IS Protocol

Support for IS-IS loop-free alternate (LFA) routes essentially adds IP fast-reroute capability for IS-IS. Junos OS precomputes multiple loop-free backup routes for all IS-IS routes. These backup routes are pre-installed in the Packet Forwarding Engine, which performs a local repair and implements the backup path when the link for a primary next hop for a particular route is no longer available. The selection of LFA is done randomly by selecting any matching LFA to progress to the given destination. This does not ensure best backup coverage available for the network. In order to choose the best LFA, Junos OS allows you to configure network-wide backup selection policies for each destination (IPv4 and IPv6) and a primary next-hop interface. These policies are evaluated based on admin-group, srlg, bandwidth, protection-type, metric, and neighbor information.

During backup shortest-path-first (SPF) computation, each node and link attribute of the backup path is accumulated by IGP and is associated with every node (router) in the topology. The next hop in the best backup path is selected as the backup next hop in the routing table. In general, backup evaluation policy rules are categorized into the following types:

- Pruning — Rules configured to select the eligible backup path.
- Ordering — Rules configured to select the best among the eligible backup paths.

The backup selection policies can be configured with both pruning and ordering rules. While evaluating the backup policies, each backup path is assigned a score, an integer value that signifies the total weight of the evaluated criteria. The backup path with the highest score is selected.

To enforce LFA selection, configure various rules for the following attributes:

- admin-group— Administrative groups, also known as link coloring or resource class, are manually assigned attributes that describe the “color” of links, such that links with the same color conceptually belong to the same class. These configured administrative groups are defined under protocol MPLS. You can use administrative groups to implement a variety of backup selection policies using exclude, include-all, include-any, or preference.
- backup-neighbor— A neighbor ID to either prefer or exclude in the backup path selection.

- **node**— A list of loop-back IP addresses of the adjacent nodes to either prefer or exclude in the backup path selection. The node can be a local (adjacent router) node, remote node, or any other router in the backup path. The nodes are identified through the TE-router-ID TLV advertised by a node in the LSP.
- **node-tag**— A node tag identifies a group of nodes in the network based on criteria such as the same neighbor tag values for all PE nodes to either prefer or exclude in the a backup path selection. This is implemented using IS-IS admin-tags. The routers are not identified with the explicit router-id but with an admin-tag prefix to their lo0 address prefix. These tags are advertised as part of extended IP reachability with a /32 prefix length that represents the TE-router \_ID or node-ID of a router.
- **srlg**— A shared risk link group (SRLG) is a set of links sharing a common resource, which affects all links in the set if the common resource fails. These links share the same risk of failure and are therefore considered to belong to the same SRLG. For example, links sharing a common fiber are said to be in the same SRLG because a fault with the fiber might cause all links in the group to fail. An SRLG is represented by a 32-bit number unique within an IGP (IS-IS) domain. A link might belong to multiple SRLGs. You can define the backup selection to either allow or reject the common SRLGs between the primary and the backup path.
- **bandwidth**—The bandwidth specifies the bandwidth constraints between the primary and the backup path. The backup next-hop link can be used only if the bandwidth of the backup next-hop interface is greater than or equal to the bandwidth of the primary next hop.
- **protection-type**— The protection-type protects the destination from node failure of the primary node or link failure of the primary link. You can configure node, link, or node-link to protect the destination.. If link-node is configured , then the node-protecting LFA is preferred over link-protection LFA.
- **metric**— Metric decides how the LFAs should be preferred. In backup selection path, root metric and dest-metric are the two types of metrics. root-metric indicates the metric to the one-hop neighbor or a remote router such as an RSVP backup LSP tail-end router. The dest-metric indicates the metric from a one-hop neighbor or remote router such as an RSVP backup LSP tail-end router to the final destination. The metric evaluation is done either in ascending or descending order. By default, the first preference is given to backup paths with lowest destination evaluation and then to backup paths with lowest root metrics.

The evaluation-order allows you to control the order and criteria of evaluating these attributes in the backup path. You can explicitly configure the evaluation order. Only the configured attributes influence the backup path selection. The default order of evaluation of these attributes for the LFA is [ admin-group srlg bandwidth protection-type neighbor neighbor-tag metric ] .

## RELATED DOCUMENTATION

[Example: Configuring Backup Selection Policy for IS-IS Protocol](#)

## Example: Configuring Backup Selection Policy for IS-IS Protocol

### IN THIS SECTION

- [Requirements | 105](#)
- [Overview | 105](#)
- [Configuration | 107](#)
- [Verification | 133](#)

This example shows how to configure the backup selection policy for the IS-IS protocol.

When you enable backup selection policies, Junos OS allows selection of LFA based on the policy rules and attributes of the links and nodes in the network. These attributes are admin-group, srlg, bandwidth, protection-type, metric, neighbor, and neighbor-tag.

### Requirements

This example uses the following hardware and software components:

- Eight routers that can be a combination of Edge Routers and Core Routers.
- Junos OS running on all devices.

Before you begin:

1. Configure the device interfaces.
2. Configure IS-IS.

### Overview

#### IN THIS SECTION

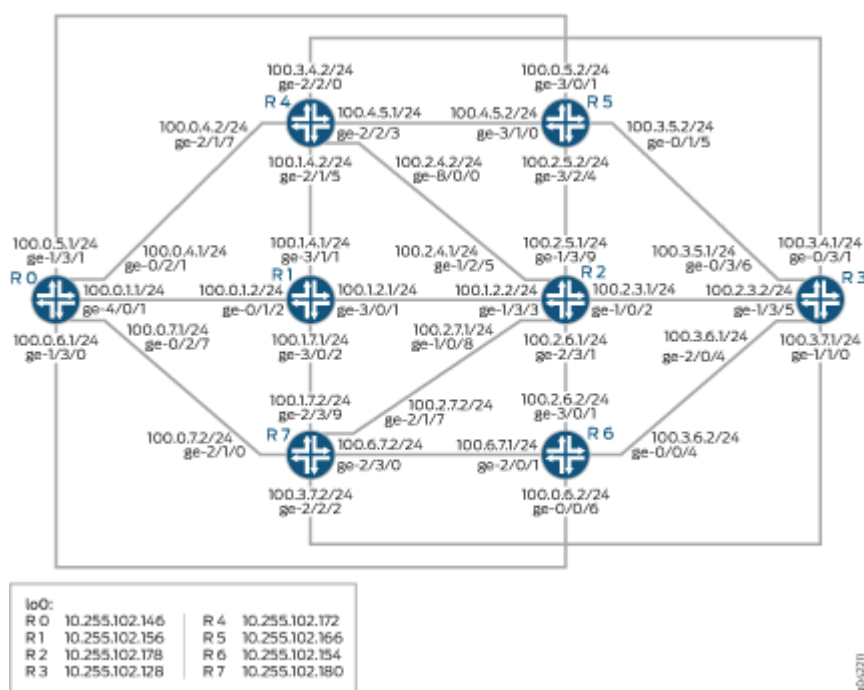
- [Topology | 106](#)

The default loop free alternative (LFA) selection algorithm or criteria can be overridden with an LFA policy. These policies are configured for each destination (IPv4 and IPv6) and a primary next-hop interface. These backup policies enforce LFA selection based on admin-group, srlg, bandwidth, protection-type, metric, neighbor, and neighbor-tag attributes of the backup path. During backup shortest-path-first (SPF) computation, each attribute (both node and link) of the backup path, stored per backup next-hop, is accumulated by IGP. For the routes created internally by IGP, the attribute set of every backup path is evaluated against the policy configured for each destination (IPv4 and IPv6) and a primary next-hop interface. The first or the best backup path is selected and installed as the backup next hop in the routing table. To configure the backup selection policy, include the backup-selection configuration statement at the [edit routing-options] hierarchy level. The show backup-selection command displays the configured policies for a given interface and destination. The display can be filtered against a particular destination, prefix, interface, or logical systems.

## Topology

In this topology, backup selection policy is configured on Device R3.

Figure 11: Backup Selection Path





## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 107](#)
- [Configuring Device R3 | 123](#)
- [Results | 128](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### RO

```
set interfaces ge-4/0/1 unit 0 family inet address 100.0.1.1/24
set interfaces ge-4/0/1 unit 0 family iso
set interfaces ge-4/0/1 unit 0 family inet6 address 2001:100:0:1::1/64
set interfaces ge-4/0/1 unit 0 family mpls
set interfaces ge-0/2/1 unit 0 family inet address 100.0.4.1/24
set interfaces ge-0/2/1 unit 0 family iso
set interfaces ge-0/2/1 unit 0 family inet6 address 2001:100:0:4::1/64
set interfaces ge-0/2/1 unit 0 family mpls
set interfaces ge-1/3/1 unit 0 family inet address 100.0.5.1/24
set interfaces ge-1/3/1 unit 0 family iso
set interfaces ge-1/3/1 unit 0 family inet6 address 2001:100:0:5::1/64
set interfaces ge-1/3/1 unit 0 family mpls
set interfaces ge-1/3/0 unit 0 family inet address 100.0.6.1/24
set interfaces ge-1/3/0 unit 0 family iso
set interfaces ge-1/3/0 unit 0 family inet6 address 2001:100:0:6::1/64
set interfaces ge-1/3/0 unit 0 family mpls
set interfaces ge-0/2/7 unit 0 family inet address 100.0.7.1/24
set interfaces ge-0/2/7 unit 0 family iso
set interfaces ge-0/2/7 unit 0 family inet6 address 2001:100:0:7::1/64
set interfaces ge-0/2/7 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.102.146/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0100.1001.00
set interfaces lo0 unit 0 family inet6 address abcd::10:255:102:146/128
```

```
set interfaces lo0 unit 0 family mpls
set protocols rsvp interface all
set protocols mpls admin-groups c0 0
set protocols mpls admin-groups c1 1
set protocols mpls admin-groups c2 2
set protocols mpls admin-groups c3 3
set protocols mpls admin-groups c4 4
set protocols mpls admin-groups c5 5
set protocols mpls admin-groups c6 6
set protocols mpls admin-groups c7 7
set protocols mpls admin-groups c8 8
set protocols mpls admin-groups c9 9
set protocols mpls admin-groups c10 10
set protocols mpls admin-groups c11 11
set protocols mpls admin-groups c12 12
set protocols mpls admin-groups c13 13
set protocols mpls admin-groups c14 14
set protocols mpls admin-groups c15 15
set protocols mpls admin-groups c16 16
set protocols mpls admin-groups c17 17
set protocols mpls admin-groups c18 18
set protocols mpls admin-groups c19 19
set protocols mpls admin-groups c20 20
set protocols mpls admin-groups c21 21
set protocols mpls admin-groups c22 22
set protocols mpls admin-groups c23 23
set protocols mpls admin-groups c24 24
set protocols mpls admin-groups c25 25
set protocols mpls admin-groups c26 26
set protocols mpls admin-groups c27 27
set protocols mpls admin-groups c28 28
set protocols mpls admin-groups c29 29
set protocols mpls admin-groups c30 30
set protocols mpls admin-groups c31 31
set protocols mpls interface all
set protocols isis interface ge-4/0/1 level 2 metric 10
set protocols isis interface ge-0/2/1 level 2 metric 18
set protocols isis interface ge-1/3/1 level 2 metric 51
set protocols isis interface ge-1/3/0 level 2 metric 52
set protocols isis interface ge-0/2/7 level 2 metric 23
set protocols isis interface all level 2 metric 10
set routing-options srlg srlg1 srlg-value 101
set routing-options srlg srlg2 srlg-value 102
```

```

set routing-options srlg srlg3 srlg-value 103
set routing-options srlg srlg4 srlg-value 104
set routing-options srlg srlg5 srlg-value 105
set routing-options srlg srlg6 srlg-value 106
set routing-options srlg srlg7 srlg-value 107
set routing-options srlg srlg8 srlg-value 108
set routing-options srlg srlg9 srlg-value 109
set routing-options srlg srlg10 srlg-value 110
set routing-options srlg srlg11 srlg-value 111
set routing-options srlg srlg12 srlg-value 112

```

## R1

```

set interfaces ge-0/1/2 unit 0 family inet address 100.0.1.2/24
set interfaces ge-0/1/2 unit 0 family iso
set interfaces ge-0/1/2 unit 0 family inet6 address 2001:100:0:1::2/64
set interfaces ge-0/1/2 unit 0 family mpls
set interfaces ge-3/0/1 unit 0 family inet address 100.1.2.1/24
set interfaces ge-3/0/1 unit 0 family iso
set interfaces ge-3/0/1 unit 0 family inet6 address 2001:100:1:2::1/64
set interfaces ge-3/0/1 unit 0 family mpls
set interfaces ge-3/1/1 unit 0 family inet address 100.1.4.1/24
set interfaces ge-3/1/1 unit 0 family iso
set interfaces ge-3/1/1 unit 0 family inet6 address 2001:100:1:4::1/64
set interfaces ge-3/1/1 unit 0 family mpls
set interfaces ge-3/0/2 unit 0 family inet address 100.1.7.1/24
set interfaces ge-3/0/2 unit 0 family iso
set interfaces ge-3/0/2 unit 0 family inet6 address 2001:100:1:7::1/64
set interfaces ge-3/0/2 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.102.156/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0100.1002.00
set interfaces lo0 unit 0 family inet6 address abcd::10:255:102:156/128
set interfaces lo0 unit 0 family mpls
set protocols rsvp interface all
set protocols mpls admin-groups c0 0
set protocols mpls admin-groups c1 1
set protocols mpls admin-groups c2 2
set protocols mpls admin-groups c3 3
set protocols mpls admin-groups c4 4
set protocols mpls admin-groups c5 5
set protocols mpls admin-groups c6 6
set protocols mpls admin-groups c7 7

```

```
set protocols mpls admin-groups c8 8
set protocols mpls admin-groups c9 9
set protocols mpls admin-groups c10 10
set protocols mpls admin-groups c11 11
set protocols mpls admin-groups c12 12
set protocols mpls admin-groups c13 13
set protocols mpls admin-groups c14 14
set protocols mpls admin-groups c15 15
set protocols mpls admin-groups c16 16
set protocols mpls admin-groups c17 17
set protocols mpls admin-groups c18 18
set protocols mpls admin-groups c19 19
set protocols mpls admin-groups c20 20
set protocols mpls admin-groups c21 21
set protocols mpls admin-groups c22 22
set protocols mpls admin-groups c23 23
set protocols mpls admin-groups c24 24
set protocols mpls admin-groups c25 25
set protocols mpls admin-groups c26 26
set protocols mpls admin-groups c27 27
set protocols mpls admin-groups c28 28
set protocols mpls admin-groups c29 29
set protocols mpls admin-groups c30 30
set protocols mpls admin-groups c31 31
set protocols mpls interface all
set protocols mpls interface ge-0/1/2 srlg srlg9
set protocols mpls interface ge-0/1/2 admin-group c1
set protocols mpls interface ge-0/1/2 admin-group c2
set protocols mpls interface ge-0/1/2 admin-group c6
set protocols isis interface all level 2 metric 10
set routing-options srlg srlg1 srlg-value 101
set routing-options srlg srlg2 srlg-value 102
set routing-options srlg srlg3 srlg-value 103
set routing-options srlg srlg4 srlg-value 104
set routing-options srlg srlg5 srlg-value 105
set routing-options srlg srlg6 srlg-value 106
set routing-options srlg srlg7 srlg-value 107
set routing-options srlg srlg8 srlg-value 108
set routing-options srlg srlg9 srlg-value 109
set routing-options srlg srlg10 srlg-value 110
set routing-options srlg srlg11 srlg-value 111
set routing-options srlg srlg12 srlg-value 112
```

R2

```
set interfaces ge-1/3/3 unit 0 family inet address 100.1.2.2/24
set interfaces ge-1/3/3 unit 0 family iso
set interfaces ge-1/3/3 unit 0 family inet6 address 2001:100:1:2::2/64
set interfaces ge-1/3/3 unit 0 family mpls
set interfaces ge-1/0/2 unit 0 family inet address 100.2.3.1/24
set interfaces ge-1/0/2 unit 0 family iso
set interfaces ge-1/0/2 unit 0 family inet6 address 2001:100:2:3::1/64
set interfaces ge-1/0/2 unit 0 family mpls
set interfaces ge-1/2/5 unit 0 family inet address 100.2.4.1/24
set interfaces ge-1/2/5 unit 0 family iso
set interfaces ge-1/2/5 unit 0 family inet6 address 2001:100:2:4::1/64
set interfaces ge-1/2/5 unit 0 family mpls
set interfaces ge-1/3/9 unit 0 family inet address 100.2.5.1/24
set interfaces ge-1/3/9 unit 0 family iso
set interfaces ge-1/3/9 unit 0 family inet6 address 2001:100:2:5::1/64
set interfaces ge-2/3/1 unit 0 family inet address 100.2.6.1/24
set interfaces ge-2/3/1 unit 0 family iso
set interfaces ge-2/3/1 unit 0 family inet6 address 2001:100:2:6::1/64
set interfaces ge-2/3/1 unit 0 family mpls
set interfaces ge-1/0/8 unit 0 family inet address 100.2.7.1/24
set interfaces ge-1/0/8 unit 0 family iso
set interfaces ge-1/0/8 unit 0 family inet6 address 2001:100:2:7::1/64
set interfaces ge-1/0/8 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.102.178/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0100.1003.00
set interfaces lo0 unit 0 family inet6 address abcd::10:255:102:178/128
set interfaces lo0 unit 0 family mpls
set protocols rsvp interface all
set protocols mpls admin-groups c0 0
set protocols mpls admin-groups c1 1
set protocols mpls admin-groups c2 2
set protocols mpls admin-groups c3 3
set protocols mpls admin-groups c4 4
set protocols mpls admin-groups c5 5
set protocols mpls admin-groups c6 6
set protocols mpls admin-groups c7 7
set protocols mpls admin-groups c8 8
set protocols mpls admin-groups c9 9
set protocols mpls admin-groups c10 10
set protocols mpls admin-groups c11 11
```

```
set protocols mpls admin-groups c12 12
set protocols mpls admin-groups c13 13
set protocols mpls admin-groups c14 14
set protocols mpls admin-groups c15 15
set protocols mpls admin-groups c16 16
set protocols mpls admin-groups c17 17
set protocols mpls admin-groups c18 18
set protocols mpls admin-groups c19 19
set protocols mpls admin-groups c20 20
set protocols mpls admin-groups c21 21
set protocols mpls admin-groups c22 22
set protocols mpls admin-groups c23 23
set protocols mpls admin-groups c24 24
set protocols mpls admin-groups c25 25
set protocols mpls admin-groups c26 26
set protocols mpls admin-groups c27 27
set protocols mpls admin-groups c28 28
set protocols mpls admin-groups c29 29
set protocols mpls admin-groups c30 30
set protocols mpls admin-groups c31 31
set protocols mpls interface all
set protocols mpls interface ge-1/0/2 srlg srlg1
set protocols mpls interface ge-1/3/9 srlg srlg1
set protocols mpls interface ge-1/3/9 srlg srlg2
set protocols mpls interface ge-2/3/1 srlg srlg1
set protocols mpls interface ge-1/0/8 srlg srlg7
set protocols isis interface ge-1/0/2 link-protection
set protocols isis interface ge-1/2/5 level 2 metric 12
set protocols isis interface ge-2/3/1 level 2 metric 12
set protocols isis interface ge-1/0/8 level 2 metric 13
set protocols isis interface all level 2 metric 10
set routing-options srlg srlg1 srlg-value 101
set routing-options srlg srlg2 srlg-value 102
set routing-options srlg srlg3 srlg-value 103
set routing-options srlg srlg4 srlg-value 104
set routing-options srlg srlg5 srlg-value 105
set routing-options srlg srlg6 srlg-value 106
set routing-options srlg srlg7 srlg-value 107
set routing-options srlg srlg8 srlg-value 108
set routing-options srlg srlg9 srlg-value 109
set routing-options srlg srlg10 srlg-value 110
```

```
set routing-options srlg srlg11 srlg-value 111
set routing-options srlg srlg12 srlg-value 112
```

### R3

```
set interfaces ge-1/3/5 unit 0 family inet address 100.2.3.2/24
set interfaces ge-1/3/5 unit 0 family iso
set interfaces ge-1/3/5 unit 0 family inet6 address 2001:100:2:3::2/64
set interfaces ge-1/3/5 unit 0 family mpls
set interfaces ge-0/3/1 unit 0 family inet address 100.3.4.1/24
set interfaces ge-0/3/1 unit 0 family iso
set interfaces ge-0/3/1 unit 0 family inet6 address 2001:100:3:4::1/64
set interfaces ge-0/3/1 unit 0 family mpls
set interfaces ge-0/3/6 unit 0 family inet address 100.3.5.1/24
set interfaces ge-0/3/6 unit 0 family iso
set interfaces ge-0/3/6 unit 0 family inet6 address 2001:100:3:5::1/64
set interfaces ge-0/3/6 unit 0 family mpls
set interfaces ge-2/0/4 unit 0 family inet address 100.3.6.1/24
set interfaces ge-2/0/4 unit 0 family iso
set interfaces ge-2/0/4 unit 0 family inet6 address 2001:100:3:6::1/64
set interfaces ge-2/0/4 unit 0 family mpls
set interfaces ge-1/1/0 unit 0 family inet address 100.3.7.1/24
set interfaces ge-1/1/0 unit 0 family iso
set interfaces ge-1/1/0 unit 0 family inet6 address 2001:100:3:7::1/64
set interfaces ge-1/1/0 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.102.128/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0100.1004.00
set interfaces lo0 unit 0 family inet6 address abcd::10:255:102:128/128
set interfaces lo0 unit 0 family mpls
set policy-options policy-statement ecmp term 1 then load-balance per-packet
set protocols rsvp interface all
set protocols mpls admin-groups c0 0
set protocols mpls admin-groups c1 1
set protocols mpls admin-groups c2 2
set protocols mpls admin-groups c3 3
set protocols mpls admin-groups c4 4
set protocols mpls admin-groups c5 5
set protocols mpls admin-groups c6 6
set protocols mpls admin-groups c7 7
set protocols mpls admin-groups c8 8
set protocols mpls admin-groups c9 9
set protocols mpls admin-groups c10 10
```

```
set protocols mpls admin-groups c11 11
set protocols mpls admin-groups c12 12
set protocols mpls admin-groups c13 13
set protocols mpls admin-groups c14 14
set protocols mpls admin-groups c15 15
set protocols mpls admin-groups c16 16
set protocols mpls admin-groups c17 17
set protocols mpls admin-groups c18 18
set protocols mpls admin-groups c19 19
set protocols mpls admin-groups c20 20
set protocols mpls admin-groups c21 21
set protocols mpls admin-groups c22 22
set protocols mpls admin-groups c23 23
set protocols mpls admin-groups c24 24
set protocols mpls admin-groups c25 25
set protocols mpls admin-groups c26 26
set protocols mpls admin-groups c27 27
set protocols mpls admin-groups c28 28
set protocols mpls admin-groups c29 29
set protocols mpls admin-groups c30 30
set protocols mpls admin-groups c31 31
set protocols mpls interface all
set protocols mpls interface ge-0/3/1 srlg srlg1
set protocols mpls interface ge-0/3/1 srlg srlg2
set protocols mpls interface ge-0/3/1 admin-group c1
set protocols mpls interface ge-0/3/1 admin-group c2
set protocols mpls interface ge-0/3/1 admin-group c3
set protocols mpls interface ge-0/3/1 admin-group c5
set protocols mpls interface ge-0/3/6 admin-group c1
set protocols mpls interface ge-0/3/6 admin-group c2
set protocols mpls interface ge-2/0/4 admin-group c1
set protocols mpls interface ge-2/0/4 admin-group c2
set protocols mpls interface ge-2/0/4 admin-group c5
set protocols mpls interface ge-1/1/0 admin-group c2
set protocols mpls interface ge-1/1/0 admin-group c12
set protocols isis interface ge-1/3/5 link-protection
set protocols isis interface ge-0/3/1 level 2 metric 21
set protocols isis interface ge-0/3/6 level 2 metric 13
set protocols isis interface ge-2/0/4 level 2 metric 15
set protocols isis interface ge-1/1/0 level 2 metric 22
set protocols isis interface all level 2 metric 10
set routing-options forwarding-table export ecmp
set routing-options srlg srlg1 srlg-value 101
```



```

set routing-options srlg srlg2 srlg-value 102
set routing-options srlg srlg3 srlg-value 103
set routing-options srlg srlg4 srlg-value 104
set routing-options srlg srlg5 srlg-value 105
set routing-options srlg srlg6 srlg-value 106
set routing-options srlg srlg7 srlg-value 107
set routing-options srlg srlg8 srlg-value 108
set routing-options srlg srlg9 srlg-value 109
set routing-options srlg srlg10 srlg-value 110
set routing-options srlg srlg11 srlg-value 111
set routing-options srlg srlg12 srlg-value 112
set routing-options backup-selection destination 0.0.0.0/0 interface all admin-group include-
all c1
set routing-options backup-selection destination 0.0.0.0/0 interface all admin-group include-
any c2
set routing-options backup-selection destination 0.0.0.0/0 interface all admin-group preference
c3
set routing-options backup-selection destination 0.0.0.0/0 interface all srlg loose
set routing-options backup-selection destination 0.0.0.0/0 interface all downstream-paths-only
set routing-options backup-selection destination 0.0.0.0/0 interface all bandwidth-greater-
equal-primary
set routing-options backup-selection destination 0.0.0.0/0 interface all neighbor preference
10.255.102.178
set routing-options backup-selection destination 0.0.0.0/0 interface all neighbor-tag
preference 1004
set routing-options backup-selection destination 0.0.0.0/0 interface all metric-order dest
set routing-options backup-selection destination 0.0.0.0/0 interface all evaluation-order admin-
group
set routing-options backup-selection destination 0.0.0.0/0 interface all evaluation-order srlg
set routing-options backup-selection destination 0.0.0.0/0 interface all evaluation-order
bandwidth
set routing-options backup-selection destination 100.0.1.0/24 interface all srlg strict
set routing-options backup-selection destination 100.0.1.0/24 interface all bandwidth-greater-
equal-primary
set routing-options backup-selection destination 100.0.7.0/24 interface all srlg strict

```

#### R4

```

set interfaces ge-2/1/7 unit 0 family inet address 100.0.4.2/24
set interfaces ge-2/1/7 unit 0 family iso
set interfaces ge-2/1/7 unit 0 family inet6 address 2001:100:0:4::2/64
set interfaces ge-2/1/7 unit 0 family mpls

```

```

set interfaces ge-2/1/5 unit 0 family inet address 100.1.4.2/24
set interfaces ge-2/1/5 unit 0 family iso
set interfaces ge-2/1/5 unit 0 family inet6 address 2001:100:1:4::2/64
set interfaces ge-2/1/5 unit 0 family mpls
set interfaces ge-8/0/0 unit 0 family inet address 100.2.4.2/24
set interfaces ge-8/0/0 unit 0 family iso
set interfaces ge-8/0/0 unit 0 family inet6 address 2001:100:2:4::2/64
set interfaces ge-8/0/0 unit 0 family mpls
set interfaces ge-2/2/0 unit 0 family inet address 100.3.4.2/24
set interfaces ge-2/2/0 unit 0 family iso
set interfaces ge-2/2/0 unit 0 family inet6 address 2001:100:3:4::2/64
set interfaces ge-2/2/0 unit 0 family mpls
set interfaces ge-2/2/3 unit 0 family inet address 100.4.5.1/24
set interfaces ge-2/2/3 unit 0 family iso
set interfaces ge-2/2/3 unit 0 family inet6 address 2001:100:4:5::1/64
set interfaces ge-2/2/3 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.102.172/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0100.1005.00
set interfaces lo0 unit 0 family inet6 address abcd::10:255:102:172/128
set interfaces lo0 unit 0 family mpls
set protocols rsvp interface all
set protocols mpls admin-groups c0 0
set protocols mpls admin-groups c1 1
set protocols mpls admin-groups c2 2
set protocols mpls admin-groups c3 3
set protocols mpls admin-groups c4 4
set protocols mpls admin-groups c5 5
set protocols mpls admin-groups c6 6
set protocols mpls admin-groups c7 7
set protocols mpls admin-groups c8 8
set protocols mpls admin-groups c9 9
set protocols mpls admin-groups c10 10
set protocols mpls admin-groups c11 11
set protocols mpls admin-groups c12 12
set protocols mpls admin-groups c13 13
set protocols mpls admin-groups c14 14
set protocols mpls admin-groups c15 15
set protocols mpls admin-groups c16 16
set protocols mpls admin-groups c17 17
set protocols mpls admin-groups c18 18
set protocols mpls admin-groups c19 19
set protocols mpls admin-groups c20 20
set protocols mpls admin-groups c21 21

```

```

set protocols mpls admin-groups c22 22
set protocols mpls admin-groups c23 23
set protocols mpls admin-groups c24 24
set protocols mpls admin-groups c25 25
set protocols mpls admin-groups c26 26
set protocols mpls admin-groups c27 27
set protocols mpls admin-groups c28 28
set protocols mpls admin-groups c29 29
set protocols mpls admin-groups c30 30
set protocols mpls admin-groups c31 31
set protocols mpls interface all
set protocols mpls interface ge-2/1/7 srlg srlg1
set protocols mpls interface ge-2/1/7 srlg srlg2
set protocols mpls interface ge-2/1/7 srlg srlg3
set protocols mpls interface ge-2/1/7 admin-group c1
set protocols mpls interface ge-2/1/7 admin-group c2
set protocols mpls interface ge-2/1/7 admin-group c6
set protocols mpls interface ge-2/1/7 admin-group c13
set protocols isis interface ge-2/1/7 level 2 metric 18
set protocols isis interface ge-8/0/0 level 2 metric 12
set protocols isis interface ge-2/2/0 level 2 metric 21
set protocols isis interface ge-2/2/3 level 2 metric 10
set protocols isis interface all level 2 metric 10
set routing-options srlg srlg1 srlg-value 101
set routing-options srlg srlg2 srlg-value 102
set routing-options srlg srlg3 srlg-value 103
set routing-options srlg srlg4 srlg-value 104
set routing-options srlg srlg5 srlg-value 105
set routing-options srlg srlg6 srlg-value 106
set routing-options srlg srlg7 srlg-value 107
set routing-options srlg srlg8 srlg-value 108
set routing-options srlg srlg9 srlg-value 109
set routing-options srlg srlg10 srlg-value 110
set routing-options srlg srlg11 srlg-value 111
set routing-options srlg srlg12 srlg-value 112

```

## R5

```

set interfaces ge-3/0/1 unit 0 family inet address 100.0.5.2/24
set interfaces ge-3/0/1 unit 0 family iso
set interfaces ge-3/0/1 unit 0 family inet6 address 2001:100:0:5::2/64
set interfaces ge-3/0/1 unit 0 family mpls

```

```

set interfaces ge-3/2/4 unit 0 family inet address 100.2.5.2/24
set interfaces ge-3/2/4 unit 0 family iso
set interfaces ge-3/2/4 unit 0 family inet6 address 2001:100:2:5::2/64
set interfaces ge-3/2/4 unit 0 family mpls
set interfaces ge-0/1/5 unit 0 family inet address 100.3.5.2/24
set interfaces ge-0/1/5 unit 0 family iso
set interfaces ge-0/1/5 unit 0 family inet6 address 2001:100:3:5::2/64
set interfaces ge-0/1/5 unit 0 family mpls
set interfaces ge-3/1/0 unit 0 family inet address 100.4.5.2/24
set interfaces ge-3/1/0 unit 0 family iso
set interfaces ge-3/1/0 unit 0 family inet6 address 2001:100:4:5::2/64
set interfaces ge-3/1/0 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.102.166/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0100.1006.00
set interfaces lo0 unit 0 family inet6 address abcd::10:255:102:166/128
set interfaces lo0 unit 0 family mpls
set protocols rsvp interface all
set protocols mpls admin-groups c0 0
set protocols mpls admin-groups c1 1
set protocols mpls admin-groups c2 2
set protocols mpls admin-groups c3 3
set protocols mpls admin-groups c4 4
set protocols mpls admin-groups c5 5
set protocols mpls admin-groups c6 6
set protocols mpls admin-groups c7 7
set protocols mpls admin-groups c8 8
set protocols mpls admin-groups c9 9
set protocols mpls admin-groups c10 10
set protocols mpls admin-groups c11 11
set protocols mpls admin-groups c12 12
set protocols mpls admin-groups c13 13
set protocols mpls admin-groups c14 14
set protocols mpls admin-groups c15 15
set protocols mpls admin-groups c16 16
set protocols mpls admin-groups c17 17
set protocols mpls admin-groups c18 18
set protocols mpls admin-groups c19 19
set protocols mpls admin-groups c20 20
set protocols mpls admin-groups c21 21
set protocols mpls admin-groups c22 22
set protocols mpls admin-groups c23 23
set protocols mpls admin-groups c24 24
set protocols mpls admin-groups c25 25

```

```

set protocols mpls admin-groups c26 26
set protocols mpls admin-groups c27 27
set protocols mpls admin-groups c28 28
set protocols mpls admin-groups c29 29
set protocols mpls admin-groups c30 30
set protocols mpls admin-groups c31 31
set protocols mpls interface all
set protocols mpls interface ge-3/1/0 admin-group c1
set protocols mpls interface ge-3/1/0 admin-group c2
set protocols mpls interface ge-3/1/0 admin-group c4
set protocols mpls interface ge-3/0/1 srlg srlg3
set protocols mpls interface ge-3/0/1 srlg srlg4
set protocols isis interface ge-3/0/1 level 2 metric 51
set protocols isis interface ge-0/1/5 level 2 metric 13
set protocols isis interface ge-3/1/0 level 2 metric 10
set protocols isis interface all level 2 metric 10
set routing-options srlg srlg1 srlg-value 101
set routing-options srlg srlg2 srlg-value 102
set routing-options srlg srlg3 srlg-value 103
set routing-options srlg srlg4 srlg-value 104
set routing-options srlg srlg5 srlg-value 105
set routing-options srlg srlg6 srlg-value 106
set routing-options srlg srlg7 srlg-value 107
set routing-options srlg srlg8 srlg-value 108
set routing-options srlg srlg9 srlg-value 109
set routing-options srlg srlg10 srlg-value 110
set routing-options srlg srlg11 srlg-value 111
set routing-options srlg srlg12 srlg-value 112

```

## R6

```

set interfaces ge-0/0/6 unit 0 family inet address 100.0.6.2/24
set interfaces ge-0/0/6 unit 0 family iso
set interfaces ge-0/0/6 unit 0 family inet6 address 2001:100:0:6::2/64
set interfaces ge-0/0/6 unit 0 family mpls
set interfaces ge-3/0/1 unit 0 family inet address 100.2.6.2/24
set interfaces ge-3/0/1 unit 0 family iso
set interfaces ge-3/0/1 unit 0 family inet6 address 2001:100:2:6::2/64
set interfaces ge-3/0/1 unit 0 family mpls
set interfaces ge-0/0/4 unit 0 family inet address 100.3.6.2/24
set interfaces ge-0/0/4 unit 0 family iso
set interfaces ge-0/0/4 unit 0 family inet6 address 2001:100:3:6::2/64

```

```
set interfaces ge-0/0/4 unit 0 family mpls
set interfaces ge-2/0/1 unit 0 family inet address 100.6.7.1/24
set interfaces ge-2/0/1 unit 0 family iso
set interfaces ge-2/0/1 unit 0 family inet6 address 2001:100:6:7::1/64
set interfaces ge-2/0/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.102.154/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0100.1007.00
set interfaces lo0 unit 0 family inet6 address abcd::10:255:102:154/128
set interfaces lo0 unit 0 family mpls
set protocols rsvp interface all
set protocols mpls admin-groups c0 0
set protocols mpls admin-groups c1 1
set protocols mpls admin-groups c2 2
set protocols mpls admin-groups c3 3
set protocols mpls admin-groups c4 4
set protocols mpls admin-groups c5 5
set protocols mpls admin-groups c6 6
set protocols mpls admin-groups c7 7
set protocols mpls admin-groups c8 8
set protocols mpls admin-groups c9 9
set protocols mpls admin-groups c10 10
set protocols mpls admin-groups c11 11
set protocols mpls admin-groups c12 12
set protocols mpls admin-groups c13 13
set protocols mpls admin-groups c14 14
set protocols mpls admin-groups c15 15
set protocols mpls admin-groups c16 16
set protocols mpls admin-groups c17 17
set protocols mpls admin-groups c18 18
set protocols mpls admin-groups c19 19
set protocols mpls admin-groups c20 20
set protocols mpls admin-groups c21 21
set protocols mpls admin-groups c22 22
set protocols mpls admin-groups c23 23
set protocols mpls admin-groups c24 24
set protocols mpls admin-groups c25 25
set protocols mpls admin-groups c26 26
set protocols mpls admin-groups c27 27
set protocols mpls admin-groups c28 28
set protocols mpls admin-groups c29 29
set protocols mpls admin-groups c30 30
set protocols mpls admin-groups c31 31
set protocols mpls interface all
```

```

set protocols mpls interface ge-2/0/1 srlg srlg2
set protocols mpls interface ge-2/0/1 srlg srlg3
set protocols mpls interface ge-2/0/1 srlg srlg4
set protocols mpls interface ge-2/0/1 srlg srlg5
set protocols mpls interface ge-2/0/1 admin-group c1
set protocols mpls interface ge-2/0/1 admin-group c2
set protocols mpls interface ge-2/0/1 admin-group c5
set protocols mpls interface ge-2/0/1 admin-group c11
set protocols isis interface ge-0/0/6 level 2 metric 52
set protocols isis interface ge-3/0/1 level 2 metric 12
set protocols isis interface ge-0/0/4 level 2 metric 15
set protocols isis interface ge-2/0/1 level 2 metric 10
set protocols isis interface all level 2 metric 10
set routing-options srlg srlg1 srlg-value 101
set routing-options srlg srlg2 srlg-value 102
set routing-options srlg srlg3 srlg-value 103
set routing-options srlg srlg4 srlg-value 104
set routing-options srlg srlg5 srlg-value 105
set routing-options srlg srlg6 srlg-value 106
set routing-options srlg srlg7 srlg-value 107
set routing-options srlg srlg8 srlg-value 108
set routing-options srlg srlg9 srlg-value 109
set routing-options srlg srlg10 srlg-value 110
set routing-options srlg srlg11 srlg-value 111
set routing-options srlg srlg12 srlg-value 112

```

## R7

```

set interfaces ge-2/1/0 unit 0 family inet address 100.0.7.2/24
set interfaces ge-2/1/0 unit 0 family iso
set interfaces ge-2/1/0 unit 0 family inet6 address 2001:100:0:7::2/64
set interfaces ge-2/1/0 unit 0 family mpls
set interfaces ge-2/3/9 unit 0 family inet address 100.1.7.2/24
set interfaces ge-2/3/9 unit 0 family iso
set interfaces ge-2/3/9 unit 0 family inet6 address 2001:100:1:7::2/64
set interfaces ge-2/3/9 unit 0 family mpls
set interfaces ge-2/1/7 unit 0 family inet address 100.2.7.2/24
set interfaces ge-2/1/7 unit 0 family iso
set interfaces ge-2/1/7 unit 0 family inet6 address 2001:100:2:7::2/64
set interfaces ge-2/1/7 unit 0 family mpls
set interfaces ge-2/2/2 unit 0 family inet address 100.3.7.2/24
set interfaces ge-2/2/2 unit 0 family iso

```

```
set interfaces ge-2/2/2 unit 0 family inet6 address 2001:100:3:7::2/64
set interfaces ge-2/2/2 unit 0 family mpls
set interfaces ge-2/3/0 unit 0 family inet address 100.6.7.2/24
set interfaces ge-2/3/0 unit 0 family iso
set interfaces ge-2/3/0 unit 0 family inet6 address 2001:100:6:7::2/64
set interfaces ge-2/3/0 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.102.180/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0100.1008.00
set interfaces lo0 unit 0 family inet6 address abcd::10:255:102:180/128
set interfaces lo0 unit 0 family mpls
set protocols rsvp interface all
set protocols mpls admin-groups c0 0
set protocols mpls admin-groups c1 1
set protocols mpls admin-groups c2 2
set protocols mpls admin-groups c3 3
set protocols mpls admin-groups c4 4
set protocols mpls admin-groups c5 5
set protocols mpls admin-groups c6 6
set protocols mpls admin-groups c7 7
set protocols mpls admin-groups c8 8
set protocols mpls admin-groups c9 9
set protocols mpls admin-groups c10 10
set protocols mpls admin-groups c11 11
set protocols mpls admin-groups c12 12
set protocols mpls admin-groups c13 13
set protocols mpls admin-groups c14 14
set protocols mpls admin-groups c15 15
set protocols mpls admin-groups c16 16
set protocols mpls admin-groups c17 17
set protocols mpls admin-groups c18 18
set protocols mpls admin-groups c19 19
set protocols mpls admin-groups c20 20
set protocols mpls admin-groups c21 21
set protocols mpls admin-groups c22 22
set protocols mpls admin-groups c23 23
set protocols mpls admin-groups c24 24
set protocols mpls admin-groups c25 25
set protocols mpls admin-groups c26 26
set protocols mpls admin-groups c27 27
set protocols mpls admin-groups c28 28
set protocols mpls admin-groups c29 29
set protocols mpls admin-groups c30 30
set protocols mpls admin-groups c31 31
```



```

set protocols mpls interface all
set protocols mpls interface ge-2/3/9 srlg srlg8
set protocols mpls interface ge-2/3/9 admin-group c1
set protocols mpls interface ge-2/3/9 admin-group c2
set protocols mpls interface ge-2/3/9 admin-group c7
set protocols isis interface ge-2/1/0 level 2 metric 23
set protocols isis interface ge-2/1/7 level 2 metric 13
set protocols isis interface ge-2/2/2 level 2 metric 22
set protocols isis interface all level 2 metric 10
set routing-options srlg srlg1 srlg-value 101
set routing-options srlg srlg2 srlg-value 102
set routing-options srlg srlg3 srlg-value 103
set routing-options srlg srlg4 srlg-value 104
set routing-options srlg srlg5 srlg-value 105
set routing-options srlg srlg6 srlg-value 106
set routing-options srlg srlg7 srlg-value 107
set routing-options srlg srlg8 srlg-value 108
set routing-options srlg srlg9 srlg-value 109
set routing-options srlg srlg10 srlg-value 110
set routing-options srlg srlg11 srlg-value 111
set routing-options srlg srlg12 srlg-value 112

```

## Configuring Device R3

### Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [Junos OS CLI User Guide](#).

To configure Device R3:

1. Configure the interfaces.

```

[edit interfaces]
user@R3# set ge-1/3/5 unit 0 family inet address 100.2.3.2/24
user@R3# set ge-1/3/5 unit 0 family iso
user@R3# set ge-1/3/5 unit 0 family inet6 address 2001:100:2:3::2/64
user@R3# set ge-1/3/5 unit 0 family mpls
user@R3# set ge-0/3/1 unit 0 family inet address 100.3.4.1/24
user@R3# set ge-0/3/1 unit 0 family iso
user@R3# set ge-0/3/1 unit 0 family inet6 address 2001:100:3:4::1/64

```

```

user@R3# set ge-0/3/1 unit 0 family mpls
user@R3# set ge-0/3/6 unit 0 family inet address 100.3.5.1/24
user@R3# set ge-0/3/6 unit 0 family iso
user@R3# set ge-0/3/6 unit 0 family inet6 address 2001:100:3:5::1/64
user@R3# set ge-0/3/6 unit 0 family mpls
user@R3# set ge-2/0/4 unit 0 family inet address 100.3.6.1/24
user@R3# set ge-2/0/4 unit 0 family iso
user@R3# set ge-2/0/4 unit 0 family inet6 address 2001:100:3:6::1/64
user@R3# set ge-2/0/4 unit 0 family mpls
user@R3# set ge-1/1/0 unit 0 family inet address 100.3.7.1/24
user@R3# set ge-1/1/0 unit 0 family iso
user@R3# set ge-1/1/0 unit 0 family inet6 address 2001:100:3:7::1/64
user@R3# set ge-1/1/0 unit 0 family mpls
user@R3# set interfaces lo0 unit 0 family inet address 10.255.102.128/32
user@R3# set interfaces lo0 unit 0 family iso address 49.0001.0010.0100.1004.00
user@R3# set interfaces lo0 unit 0 family inet6 address abcd::10:255:102:128/128
user@R3# set interfaces lo0 unit 0 family mpls

```

## 2. Configure routing policy.

```

[edit policy-options]
user@R3# set policy-statement ecmp term 1 then load-balance per-packet

```

## 3. Enable RSVP on all the interfaces.

```

[edit protocols]
user@R3# set rsvp interface all

```

## 4. Configure administrative groups.

```

[edit protocols mpls]
user@R3# set admin-groups c0 0
user@R3# set admin-groups c1 1
user@R3# set admin-groups c2 2
user@R3# set admin-groups c3 3
user@R3# set admin-groups c4 4
user@R3# set admin-groups c5 5
user@R3# set admin-groups c6 6
user@R3# set admin-groups c7 7
user@R3# set admin-groups c8 8

```

```
user@R3# set admin-groups c9 9
user@R3# set admin-groups c10 10
user@R3# set admin-groups c11 11
user@R3# set admin-groups c12 12
user@R3# set admin-groups c13 13
user@R3# set admin-groups c14 14
user@R3# set admin-groups c15 15
user@R3# set admin-groups c16 16
user@R3# set admin-groups c16 16
user@R3# set admin-groups c17 17
user@R3# set admin-groups c18 18
user@R3# set admin-groups c19 19
user@R3# set admin-groups c20 20
user@R3# set admin-groups c21 21
user@R3# set admin-groups c22 22
user@R3# set admin-groups c23 23
user@R3# set admin-groups c24 24
user@R3# set admin-groups c25 25
user@R3# set admin-groups c26 26
user@R3# set admin-groups c27 27
user@R3# set admin-groups c28 28
user@R3# set admin-groups c29 29
user@R3# set admin-groups c30 30
user@R3# set admin-groups c31 31
```

5. Configure srlg values.

```
[edit routing-options]
user@R3# set srlg srlg1 srlg-value 101
user@R3# set srlg srlg2 srlg-value 102
user@R3# set srlg srlg3 srlg-value 103
user@R3# set srlg srlg4 srlg-value 104
user@R3# set srlg srlg5 srlg-value 105
user@R3# set srlg srlg6 srlg-value 106
user@R3# set srlg srlg7 srlg-value 107
user@R3# set srlg srlg8 srlg-value 108
user@R3# set srlg srlg9 srlg-value 109
user@R3# set srlg srlg10 srlg-value 110
user@R3# set srlg srlg11 srlg-value 111
user@R3# set srlg srlg12 srlg-value 112
```

6. Enable MPLS on all the interfaces.

```
[edit protocols mpls]
user@R3# set interface all
```

7. Configure srlg on the interfaces.

```
[edit protocols mpls]
user@R3# set interface ge-0/3/1 srlg srlg1
user@R3# set interface ge-0/3/1 srlg srlg2
```

8. Configure administrative groups on the interfaces.

```
[edit protocols mpls]
user@R3# set interface ge-0/3/1 admin-group c1
user@R3# set interface ge-0/3/1 admin-group c2
user@R3# set interface ge-0/3/1 admin-group c3
user@R3# set interface ge-0/3/1 admin-group c5
user@R3# set interface ge-0/3/6 admin-group c1
user@R3# set interface ge-0/3/6 admin-group c2
user@R3# set interface ge-2/0/4 admin-group c1
user@R3# set interface ge-2/0/4 admin-group c2
user@R3# set interface ge-2/0/4 admin-group c5
user@R3# set interface ge-1/1/0 admin-group c2
user@R3# set interface ge-1/1/0 admin-group c12
```

9. Enable link protection and configure metric values on the interfaces.

```
[edit protocols]
user@R3# set isis interface ge-1/3/5 link-protection
user@R3# set isis interface ge-0/3/1 level 2 metric 21
user@R3# set isis interface ge-0/3/6 level 2 metric 13
user@R3# set isis interface ge-2/0/4 level 2 metric 15
user@R3# set isis interface ge-1/1/0 level 2 metric 22
```

10. Configure the metric value on all the interfaces.

```
[edit protocols]
user@R3# set isis interface all level 2 metric 10
```

11. Apply the routing policy to all equal cost multi paths exported from the routing table to the forwarding table.

```
[edit routing-options]
user@R3# set forwarding-table export ecmp
```

12. Configure attributes of the backup selection policy.

```
[edit routing-options]
user@R3# set backup-selection destination 0.0.0.0/0 interface all admin-group include-all c1
user@R3# set backup-selection destination 0.0.0.0/0 interface all admin-group include-any c2
user@R3# set backup-selection destination 0.0.0.0/0 interface all admin-group preference c3
user@R3# set backup-selection destination 0.0.0.0/0 interface all srlg loose
user@R3# set backup-selection destination 0.0.0.0/0 interface all downstream-paths-only
user@R3# set backup-selection destination 0.0.0.0/0 interface all bandwidth-greater-equal-
primary
user@R3# set backup-selection destination 0.0.0.0/0 interface all neighbor preference
10.255.102.178
user@R3# set backup-selection destination 0.0.0.0/0 interface all neighbor-tag preference
1004
user@R3# set backup-selection destination 0.0.0.0/0 interface all metric-order dest
user@R3# set backup-selection destination 0.0.0.0/0 interface all evaluation-order admin-
group
user@R3# set backup-selection destination 0.0.0.0/0 interface all evaluation-order srlg
user@R3# set backup-selection destination 0.0.0.0/0 interface all evaluation-order bandwidth
user@R3# set backup-selection destination 100.0.1.0/24 interface all srlg strict
user@R3# set backup-selection destination 100.0.1.0/24 interface all bandwidth-greater-
equal-primary
user@R3# set backup-selection destination 100.0.7.0/24 interface all srlg strict
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R3# show interfaces
ge-1/3/5 {
  unit 0 {
    family inet {
      address 100.2.3.2/24;
    }
    family iso;
    family inet6 {
      address 2001:100:2:3::2/64;
    }
    family mpls;
  }
}
ge-0/3/1 {
  unit 0 {
    family inet {
      address 100.3.4.1/24;
    }
    family iso;
    family inet6 {
      address 2001:100:3:4::1/64;
    }
    family mpls;
  }
}
ge-0/3/6 {
  unit 0 {
    family inet {
      address 100.3.5.1/24;
    }
    family iso;
    family inet6 {
      address 2001:100:3:5::1/64;
    }
    family mpls;
  }
}
```

```

}
ge-2/0/4 {
    unit 0 {
        family inet {
            address 100.3.6.1/24;
        }
        family iso;
        family inet6{
            address 2001:100:3:6::1/64;
        }
        family mpls;
    }
}
ge-1/1/0 {
    unit 0 {
        family inet {
            address 100.3.7.1/24;
        }
        family iso;
        family inet6{
            address 2001:100:3:7::1/64;
        }
        family mpls;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.255.102.128/32;
        }
        family iso {
            address 49.0001.0010.0100.1004.00;
        }
        family inet6{
            address abcd::10:255:102:128/128;
        }
        family mpls;
    }
}

```

```
}  
}
```

```
user@R3# show protocols
```

```
  rsvp {  
    interface all;  
  }
```

```
  mpls {  
    admin-groups {
```

```
      c0 0;
```

```
      c1 1;
```

```
      c2 2;
```

```
      c3 3;
```

```
      c4 4;
```

```
      c5 5;
```

```
      c6 6;
```

```
      c7 7;
```

```
      c8 8;
```

```
      c9 9;
```

```
      c10 10;
```

```
      c11 11;
```

```
      c12 12;
```

```
      c13 13;
```

```
      c14 14;
```

```
      c15 15;
```

```
      c16 16;
```

```
      c17 17;
```

```
      c18 18;
```

```
      c19 19;
```

```
      c20 20;
```

```
      c21 21;
```

```
      c22 22;
```

```
      c23 23;
```

```
      c24 24;
```

```
      c25 25;
```

```
      c26 26;
```

```
      c27 27;
```

```
      c28 28;
```

```
      c29 29;
```

```
      c30 30;
```

```
      c31 31;
```



```

}
interface all;
    interface ge-0/3/1 {
        srlg [ srlg1 srlg2 ];
        admin-group [ c1 c2 c3 c5 ];
    }
    interface ge-0/3/6 {
        admin-group [ c1 c2 ];
    }
    interface ge-2/0/4 {
        admin-group [ c1 c2 c5 ];
    }
    interface ge-1/1/0 {
        admin-group [ c2 c12 ];
    }
isis {
    interface ge-1/3/5 {
        link-protection;
    }
    interface ge-0/3/1 {
        level 2 metric 21;
    }
    interface ge-0/3/6 {
        level 2 metric 13;
    }
    interface ge-2/0/4 {
        level 2 metric 15;
    }
    interface ge-1/1/0 {
        level 2 metric 22;
    }
    interface all {
        level 2 metric 10;
    }
}

```

```

user@R3# show routing-options
srlg {
    srlg1 {
        srlg-value 101;
    }
}

```

```
srlg2 {  
    srlg-value 102;  
}  
srlg3 {  
    srlg-value 103;  
}  
srlg4 {  
    srlg-value 104;  
}  
srlg5 {  
    srlg-value 105;  
}  
srlg6 {  
    srlg-value 106;  
}  
srlg7 {  
    srlg-value 107;  
}  
srlg8 {  
    srlg-value 108;  
}  
srlg9 {  
    srlg-value 109;  
}  
srlg10 {  
    srlg-value 110;  
}  
srlg111 {  
    srlg-value 111;  
}  
srlg112 {  
    srlg-value 112;  
}  
}  
backup-selection {  
    destination 0.0.0.0/0 {  
        interface all {  
            admin-group {  
                include-all c1;  
                include-any c2;  
                preference c3;  
            }  
            srlg loose;  
        }  
    }  
}
```

```

        downstream-paths-only;
        bandwidth-greater-equal-primary;
        neighbor {
            preference 10.255.102.178;
        }
        neighbor-tag {
            preference 1004;
        }
        metric-order dest;
        evaluation-order [ admin-group srlg bandwidth ];
    }
}
destination 100.0.1.0/24 {
    interface all {
        srlg strict;
        bandwidth-greater-equal-primary;
    }
}
destination 100.0.7.0/24 {
    interface all {
        srlg strict;
    }
}
}
}

```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Verifying the Routes | 134](#)
- [Verifying the IS-IS Route | 139](#)
- [Verifying the Backup SPF Roots for Device R3. | 141](#)
- [Verifying the Backup Selection Policy for Device R3 | 142](#)

Verify that the configuration is working properly.

## Verifying the Routes

### Purpose

Verify that the expected routes are learned.

### Action

From operational mode, run the `show route` command for the routing table.

```
user@R3> show route

inet.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.102.128/32  *[Direct/0] 1w0d 04:14:44
                  > via lo0.31
10.255.102.146/32  *[IS-IS/18] 6d 07:19:57, metric 40
                  > to 100.2.3.1 via ge-1/3/5.0
                  to 100.3.4.2 via ge-0/3/1.0
10.255.102.154/32  *[IS-IS/18] 1w0d 04:12:02, metric 25
                  > to 100.3.6.2 via ge-2/0/4.0
10.255.102.156/32  *[IS-IS/18] 06:38:55, metric 30
                  > to 100.2.3.1 via ge-1/3/5.0
                  to 100.3.6.2 via ge-2/0/4.0
10.255.102.166/32  *[IS-IS/18] 1w0d 04:11:57, metric 23
                  > to 100.3.5.2 via ge-0/3/6.0
10.255.102.172/32  *[IS-IS/18] 1w0d 04:12:03, metric 31
                  > to 100.3.4.2 via ge-0/3/1.0
10.255.102.178/32  *[IS-IS/18] 06:38:55, metric 20
                  > to 100.2.3.1 via ge-1/3/5.0
10.255.102.180/32  *[IS-IS/18] 1w0d 04:12:51, metric 32
                  > to 100.3.7.2 via ge-1/1/0.0
100.0.1.0/24       *[IS-IS/18] 1w0d 04:11:57, metric 30
                  > to 100.2.3.1 via ge-1/3/5.0
                  to 100.3.4.2 via ge-0/3/1.0
100.0.4.0/24       *[IS-IS/18] 1w0d 04:12:03, metric 39
                  > to 100.3.4.2 via ge-0/3/1.0
100.0.5.0/24       *[IS-IS/18] 1w0d 04:11:57, metric 64
                  > to 100.3.5.2 via ge-0/3/6.0
100.0.6.0/24       *[IS-IS/18] 1w0d 04:12:02, metric 67
                  > to 100.3.6.2 via ge-2/0/4.0
```

```

100.0.7.0/24      *[IS-IS/18] 1w0d 04:12:51, metric 45
                  > to 100.3.7.2 via ge-1/1/0.0
100.1.2.0/24      *[IS-IS/18] 06:38:55, metric 20
                  > to 100.2.3.1 via ge-1/3/5.0
100.1.4.0/24      *[IS-IS/18] 06:38:55, metric 30
                  > to 100.2.3.1 via ge-1/3/5.0
                  to 100.3.6.2 via ge-2/0/4.0
100.1.7.0/24      *[IS-IS/18] 06:38:55, metric 30
                  > to 100.2.3.1 via ge-1/3/5.0
                  to 100.3.6.2 via ge-2/0/4.0
100.2.3.0/24      *[Direct/0] 1w0d 04:13:11
                  > via ge-1/3/5.0
100.2.3.2/32      *[Local/0] 1w0d 04:13:11
                  Local via ge-1/3/5.0
100.2.4.0/24      *[IS-IS/18] 06:38:55, metric 22
                  > to 100.2.3.1 via ge-1/3/5.0
100.2.5.0/24      *[IS-IS/18] 06:38:55, metric 20
                  > to 100.2.3.1 via ge-1/3/5.0
100.2.6.0/24      *[IS-IS/18] 06:38:55, metric 22
                  > to 100.2.3.1 via ge-1/3/5.0
100.2.7.0/24      *[IS-IS/18] 06:38:55, metric 23
                  > to 100.2.3.1 via ge-1/3/5.0
100.3.4.0/24      *[Direct/0] 1w0d 04:13:10
                  > via ge-0/3/1.0
100.3.4.1/32      *[Local/0] 1w0d 04:13:10
                  Local via ge-0/3/1.0
100.3.5.0/24      *[Direct/0] 1w0d 04:13:10
                  > via ge-0/3/6.0
100.3.5.1/32      *[Local/0] 1w0d 04:13:10
                  Local via ge-0/3/6.0
100.3.6.0/24      *[Direct/0] 1w0d 04:13:10
                  > via ge-2/0/4.0
100.3.6.1/32      *[Local/0] 1w0d 04:13:10
                  Local via ge-2/0/4.0
100.3.7.0/24      *[Direct/0] 1w0d 04:13:10
                  > via ge-1/1/0.0
100.3.7.1/32      *[Local/0] 1w0d 04:13:10
                  Local via ge-1/1/0.0
100.4.5.0/24      *[IS-IS/18] 1w0d 04:11:57, metric 23
                  > to 100.3.5.2 via ge-0/3/6.0
100.6.7.0/24      *[IS-IS/18] 1w0d 04:12:02, metric 25
                  > to 100.3.6.2 via ge-2/0/4.0

```

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

+ = Active Route, - = Last Active, \* = Both

49.0001.0010.0100.1004/72

\*[Direct/0] 1w0d 04:14:44

> via lo0.0

mpls.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)

+ = Active Route, - = Last Active, \* = Both

0 \*[MPLS/0] 1w0d 04:14:47, metric 1

Receive

1 \*[MPLS/0] 1w0d 04:14:47, metric 1

Receive

2 \*[MPLS/0] 1w0d 04:14:47, metric 1

Receive

13 \*[MPLS/0] 1w0d 04:14:47, metric 1

Receive

inet6.0: 39 destinations, 43 routes (39 active, 0 holddown, 0 hidden)

+ = Active Route, - = Last Active, \* = Both

2001:100:0:1::/64 \*[IS-IS/18] 1w0d 04:11:57, metric 30

> to fe80::2a0:a514:0:1749 via ge-1/3/5.0

to fe80::2a0:a514:0:2b49 via ge-0/3/1.0

2001:100:0:4::/64 \*[IS-IS/18] 1w0d 04:12:03, metric 39

> to fe80::2a0:a514:0:2b49 via ge-0/3/1.0

2001:100:0:5::/64 \*[IS-IS/18] 1w0d 04:11:57, metric 64

> to fe80::2a0:a514:0:3549 via ge-0/3/6.0

2001:100:0:6::/64 \*[IS-IS/18] 1w0d 04:12:02, metric 67

> to fe80::2a0:a514:0:3f49 via ge-2/0/4.0

2001:100:0:7::/64 \*[IS-IS/18] 1w0d 04:12:51, metric 45

> to fe80::2a0:a514:0:4949 via ge-1/1/0.0

2001:100:1:2::/64 \*[IS-IS/18] 1w0d 04:11:57, metric 20

> to fe80::2a0:a514:0:1749 via ge-1/3/5.0

to fe80::2a0:a514:0:3549 via ge-0/3/6.0

2001:100:1:4::/64 \*[IS-IS/18] 1w0d 04:11:57, metric 30

> to fe80::2a0:a514:0:1749 via ge-1/3/5.0

to fe80::2a0:a514:0:2b49 via ge-0/3/1.0

2001:100:1:7::/64 \*[IS-IS/18] 1w0d 04:11:57, metric 30

> to fe80::2a0:a514:0:1749 via ge-1/3/5.0

to fe80::2a0:a514:0:2b49 via ge-0/3/1.0

2001:100:2:3::/64 \*[Direct/0] 1w0d 04:13:00

```

> via ge-1/3/5.0
2001:100:2:3::2/128*[Local/0] 1w0d 04:13:11
    Local via ge-1/3/5.0
2001:100:2:4::/64 *[IS-IS/18] 1w0d 04:11:57, metric 22
    > to fe80::2a0:a514:0:1749 via ge-1/3/5.0
    to fe80::2a0:a514:0:3549 via ge-0/3/6.0
2001:100:2:5::/64 *[IS-IS/18] 1w0d 04:11:57, metric 20
    > to fe80::2a0:a514:0:1749 via ge-1/3/5.0
    to fe80::2a0:a514:0:3549 via ge-0/3/6.0
2001:100:2:6::/64 *[IS-IS/18] 1w0d 04:11:57, metric 22
    > to fe80::2a0:a514:0:1749 via ge-1/3/5.0
    to fe80::2a0:a514:0:3549 via ge-0/3/6.0
2001:100:2:7::/64 *[IS-IS/18] 1w0d 04:11:57, metric 23
    > to fe80::2a0:a514:0:1749 via ge-1/3/5.0
    to fe80::2a0:a514:0:3549 via ge-0/3/6.0
2001:100:3:4::/64 *[Direct/0] 1w0d 04:12:59
    > via ge-0/3/1.0
2001:100:3:4::1/128*[Local/0] 1w0d 04:13:10
    Local via ge-0/3/1.0
2001:100:3:5::/64 *[Direct/0] 1w0d 04:12:59
    > via ge-0/3/6.0
2001:100:3:5::1/128*[Local/0] 1w0d 04:13:10
    Local via ge-0/3/6.0
2001:100:3:6::/64 *[Direct/0] 1w0d 04:12:59
    > via ge-2/0/4.0
2001:100:3:6::1/128*[Local/0] 1w0d 04:13:10
    Local via ge-2/0/4.0
2001:100:3:7::/64 *[Direct/0] 1w0d 04:12:59
    > via ge-1/1/0.0
2001:100:3:7::1/128*[Local/0] 1w0d 04:13:10
    Local via ge-1/1/0.0
2001:100:4:5::/64 *[IS-IS/18] 1w0d 04:11:57, metric 23
    > to fe80::2a0:a514:0:3549 via ge-0/3/6.0
2001:100:6:7::/64 *[IS-IS/18] 1w0d 04:12:02, metric 25
    > to fe80::2a0:a514:0:3f49 via ge-2/0/4.0
abcd::10:255:102:128/128
    *[Direct/0] 1w0d 04:14:43
    > via lo0.0
abcd::10:255:102:146/128
    *[IS-IS/18] 1w0d 04:11:57, metric 40
    > to fe80::2a0:a514:0:1749 via ge-1/3/5.0
    to fe80::2a0:a514:0:2b49 via ge-0/3/1.0
abcd::10:255:102:154/128

```

```

        *[IS-IS/18] 1w0d 04:12:02, metric 25
        > to fe80::2a0:a514:0:3f49 via ge-2/0/4.0
abcd::10:255:102:156/128
        *[IS-IS/18] 1w0d 04:11:57, metric 30
        > to fe80::2a0:a514:0:1749 via ge-1/3/5.0
        to fe80::2a0:a514:0:2b49 via ge-0/3/1.0
abcd::10:255:102:166/128
        *[IS-IS/18] 1w0d 04:11:57, metric 23
        > to fe80::2a0:a514:0:3549 via ge-0/3/6.0
abcd::10:255:102:172/128
        *[IS-IS/18] 1w0d 04:12:03, metric 31
        > to fe80::2a0:a514:0:2b49 via ge-0/3/1.0
abcd::10:255:102:178/128
        *[IS-IS/18] 1w0d 04:11:57, metric 20
        > to fe80::2a0:a514:0:1749 via ge-1/3/5.0
        to fe80::2a0:a514:0:3549 via ge-0/3/6.0
abcd::10:255:102:180/128
        *[IS-IS/18] 1w0d 04:12:51, metric 32
        > to fe80::2a0:a514:0:4949 via ge-1/1/0.0
fe80::/64
        *[Direct/0] 1w0d 04:13:00
        > via ge-1/3/5.0
        [Direct/0] 1w0d 04:12:59
        > via ge-0/3/1.0
        [Direct/0] 1w0d 04:12:59
        > via ge-0/3/6.0
        [Direct/0] 1w0d 04:12:59
        > via ge-2/0/4.0
        [Direct/0] 1w0d 04:12:59
        > via ge-1/1/0.0
fe80::2a0:a50f:fc64:7649/128
        *[Direct/0] 1w0d 04:14:43
        > via lo0.0
fe80::2a0:a514:0:2049/128
        *[Local/0] 1w0d 04:13:11
        Local via ge-1/3/5.0
fe80::2a0:a514:0:2249/128
        *[Local/0] 1w0d 04:13:10
        Local via ge-0/3/1.0
fe80::2a0:a514:0:2349/128
        *[Local/0] 1w0d 04:13:10
        Local via ge-0/3/6.0
fe80::2a0:a514:0:2449/128
        *[Local/0] 1w0d 04:13:10

```



```

                Local via ge-2/0/4.0
fe80::2a0:a514:0:2549/128
                *[Local/0] 1w0d 04:13:10
                Local via ge-1/1/0.0

```

## Meaning

The output shows all Device R3 routes.

## Verifying the IS-IS Route

### Purpose

Verify the routing table of IS-IS.

### Action

From operational mode, run the `show isis route` command for Device R3.

```
user@R3> show isis route
```

```

IS-IS routing table          Current version: L1: 0 L2: 5023
IPv4/IPv6 Routes
-----
Prefix           L  Version  Metric Type Interface      NH Via Backup Score
10.255.102.146/32 2    5023      40 int  ge-1/3/5.0    IPV4 R2
                  ge-0/3/1.0    IPV4 R4 0000000000200000
10.255.102.154/32 2    5023      25 int  ge-2/0/4.0    IPV4 R6
10.255.102.156/32 2    5023      30 int  ge-1/3/5.0    IPV4 R2
                  ge-2/0/4.0    IPV4 R6 0000000000000000
10.255.102.166/32 2    5023      23 int  ge-0/3/6.0    IPV4 R5
10.255.102.172/32 2    5023      31 int  ge-0/3/1.0    IPV4 R4
10.255.102.178/32 2    5023      20 int  ge-1/3/5.0    IPV4 R2
10.255.102.180/32 2    5023      32 int  ge-1/1/0.0    IPV4 R7
100.0.1.0/24      2    5023      30 int  ge-1/3/5.0    IPV4 R2
                  ge-0/3/1.0    IPV4 R4 00000000002003100
100.0.4.0/24      2    5023      39 int  ge-0/3/1.0    IPV4 R4
100.0.5.0/24      2    5023      64 int  ge-0/3/6.0    IPV4 R5
100.0.6.0/24      2    5023      67 int  ge-2/0/4.0    IPV4 R6
100.0.7.0/24      2    5023      45 int  ge-1/1/0.0    IPV4 R7
100.1.2.0/24      2    5023      20 int  ge-1/3/5.0    IPV4 R2

```

100.1.4.0/24	2	5023	30	int	ge-1/3/5.0	IPV4 R2	
					ge-2/0/4.0	IPV4 R6	0000000000000000
100.1.7.0/24	2	5023	30	int	ge-1/3/5.0	IPV4 R2	
					ge-2/0/4.0	IPV4 R6	0000000000000000
100.2.4.0/24	2	5023	22	int	ge-1/3/5.0	IPV4 R2	
100.2.5.0/24	2	5023	20	int	ge-1/3/5.0	IPV4 R2	
100.2.6.0/24	2	5023	22	int	ge-1/3/5.0	IPV4 R2	
100.2.7.0/24	2	5023	23	int	ge-1/3/5.0	IPV4 R2	
100.4.5.0/24	2	5023	23	int	ge-0/3/6.0	IPV4 R5	
100.6.7.0/24	2	5023	25	int	ge-2/0/4.0	IPV4 R6	
2001:100:0:1::/64	2	5023	30	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/1.0	IPV6 R4	
2001:100:0:4::/64	2	5023	39	int	ge-0/3/1.0	IPV6 R4	
2001:100:0:5::/64	2	5023	64	int	ge-0/3/6.0	IPV6 R5	
2001:100:0:6::/64	2	5023	67	int	ge-2/0/4.0	IPV6 R6	
2001:100:0:7::/64	2	5023	45	int	ge-1/1/0.0	IPV6 R7	
2001:100:1:2::/64	2	5023	20	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/6.0	IPV6 R5	
2001:100:1:4::/64	2	5023	30	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/1.0	IPV6 R4	
2001:100:1:7::/64	2	5023	30	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/1.0	IPV6 R4	
2001:100:2:4::/64	2	5023	22	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/6.0	IPV6 R5	
2001:100:2:5::/64	2	5023	20	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/6.0	IPV6 R5	
2001:100:2:6::/64	2	5023	22	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/6.0	IPV6 R5	
2001:100:2:7::/64	2	5023	23	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/6.0	IPV6 R5	
2001:100:4:5::/64	2	5023	23	int	ge-0/3/6.0	IPV6 R5	
2001:100:6:7::/64	2	5023	25	int	ge-2/0/4.0	IPV6 R6	
abcd::10:255:102:146/128	2	5023	40	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/1.0	IPV6 R4	
abcd::10:255:102:154/128	2	5023	25	int	ge-2/0/4.0	IPV6 R6	
abcd::10:255:102:156/128	2	5023	30	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/1.0	IPV6 R4	
abcd::10:255:102:166/128	2	5023	23	int	ge-0/3/6.0	IPV6 R5	
abcd::10:255:102:172/128	2	5023	31	int	ge-0/3/1.0	IPV6 R4	
abcd::10:255:102:178/128	2	5023	20	int	ge-1/3/5.0	IPV6 R2	
					ge-0/3/6.0	IPV6 R5	
abcd::10:255:102:180/128	2	5023	32	int	ge-1/1/0.0	IPV6 R7	

## Meaning

The output displays the routing table of IS-IS routers.

## Verifying the Backup SPF Roots for Device R3.

## Purpose

Verify the potential IS-IS backup SPF roots for Device R3.

## Action

From operational mode, run the `show isis backup spf results` command for Device R3.

```
user@R3> show isis backup spf results
```

```
IS-IS level 1 SPF results:
```

```
0 nodes
```

```
IS-IS level 2 SPF results:
```

```
R0.00
```

```
Primary next-hop: ge-1/3/5, IPV4, R2, SNPA: b0:c6:9a:2a:f4:bc
```

```
Primary next-hop: ge-1/3/5, IPV6, R2, SNPA: b0:c6:9a:2a:f4:bc
```

```
Root: R4, Root Metric: 21, Metric: 18, Root Preference: 0x0
```

```
track-item: R4.04-00, track-item-isnbr: R4.00
```

```
track-item: R0.00-00, track-item-isnbr: R6.03
```

```
Eligible, Backup next-hop: ge-0/3/1, IPV4, R4, SNPA: b0:c6:9a:2a:f4:bd
```

```
Eligible, Backup next-hop: ge-0/3/1, IPV6, R4, SNPA: b0:c6:9a:2a:f4:bd
```

```
Root: R2, Root Metric: 10, Metric: 20, Root Preference: 0x0
```

```
track-item: R2.02-00, track-item-isnbr: R2.00
```

```
track-item: R1.02-00
```

```
track-item: R1.00-00, track-item-isnbr: R2.02
```

```
track-item: R0.00-00, track-item-isnbr: R1.02
```

```
Not eligible, IPV4, Reason: Primary next-hop link fate sharing
```

```
Not eligible, IPV6, Reason: Primary next-hop link fate sharing
```

```
Root: R7, Root Metric: 22, Metric: 20, Root Preference: 0x0
```

```
track-item: R7.05-00, track-item-isnbr: R7.00
```

```
track-item: R1.02-00
```

```
track-item: R1.00-00
```

```
track-item: R0.00-00, track-item-isnbr: R1.02
```

```
Eligible, Backup next-hop: ge-1/1/0, IPV4, R7, SNPA: b0:c6:9a:2a:f4:bd
```

```
Eligible, Backup next-hop: ge-1/1/0, IPV6, R7, SNPA: b0:c6:9a:2a:f4:bd
```

```

Root: R5, Root Metric: 13, Metric: 28, Root Preference: 0x0
  track-item: R5.03-00, track-item-isnbr: R5.00
  track-item: R4.04-00, track-item-isnbr: R4.00
  track-item: R4.00-00
  track-item: R0.00-00, track-item-isnbr: R6.03
  Eligible, Backup next-hop: ge-0/3/6, IPV4, R5, SNPA: b0:c6:9a:2a:f4:bd
  Eligible, Backup next-hop: ge-0/3/6, IPV6, R5, SNPA: b0:c6:9a:2a:f4:bd
Root: R6, Root Metric: 15, Metric: 30, Root Preference: 0x0
  track-item: R7.05-00, track-item-isnbr: R7.00
  track-item: R7.04-00, track-item-isnbr: R6.00
  track-item: R7.00-00
  track-item: R1.02-00
  track-item: R1.00-00
  track-item: R0.00-00, track-item-isnbr: R1.02
  Eligible, Backup next-hop: ge-2/0/4, IPV4, R6, SNPA: b0:c6:9a:2a:f4:bd
  Eligible, Backup next-hop: ge-2/0/4, IPV6, R6, SNPA: b0:c6:9a:2a:f4:bd
R7.00
Primary next-hop: ge-1/1/0, IPV4, R7, SNPA: b0:c6:9a:2a:f4:bd
Primary next-hop: ge-1/1/0, IPV6, R7, SNPA: b0:c6:9a:2a:f4:bd
Root: R7, Root Metric: 22, Metric: 0, Root Preference: 0x0
  Not eligible, IPV4, Reason: Interface protection not configured
  Not eligible, IPV6, Reason: Interface protection not configured
Root: R6, Root Metric: 15, Metric: 10, Root Preference: 0x0
  track-item: R7.04-00, track-item-isnbr: R6.00
  track-item: R7.00-00
. . .

```

## Meaning

The output displays the root calculations through each directly connected router.

## Verifying the Backup Selection Policy for Device R3

### Purpose

## Action

From operational mode, run the `show backup-selection` command for Device R3.

```
user@R3> show backup-selection

Prefix: 0.0.0.0/0
  Interface: all
    Admin-group include-all: c1
    Admin-group include-any: c2
    Admin-group preference: c3
    Neighbor preference: 10.255.102.178
    Neighbor-tag preference: 1004
    Protection Type: Link, Downstream Paths Only: Enabled, SRLG: Loose, B/w >= Primary: Enabled,
Root-metric: lowest, Dest-metric: lowest
    Metric Evaluation Order: Dest-metric, Root-metric
    Policy Evaluation Order: Admin-group, SRLG, Bandwidth
Prefix: 100.0.1.0/24
  Interface: all
    Protection Type: Link, Downstream Paths Only: Disabled, SRLG: Strict, B/w >= Primary:
Enabled, Root-metric: lowest, Dest-metric: lowest
    Metric Evaluation Order: Dest-metric, Root-metric
    Policy Evaluation Order: Admin-group, SRLG, Bandwidth, Protection, Neighbor, Metric,
Neighbor-Tag
Prefix: 100.0.7.0/24
  Interface: all
    Protection Type: Link, Downstream Paths Only: Disabled, SRLG: Strict, B/w >= Primary:
Disabled, Root-metric: lowest, Dest-metric: lowest
    Metric Evaluation Order: Dest-metric, Root-metric
    Policy Evaluation Order: Admin-group, SRLG, Bandwidth, Protection, Neighbor, Metric,
Neighbor-Tag
```

## Meaning

The output displays the configured policies per prefix per primary next-hop interface.

## RELATED DOCUMENTATION

*Understanding Backup Selection Policy for IS-IS Protocol*

*backup-selection (Protocols IS-IS)*

## Configuring Backup Selection Policy for the IS-IS Protocol

Support for IS-IS loop-free alternate (LFA) routes essentially adds IP fast-reroute capability for IS-IS. Junos OS precomputes multiple loop-free backup routes for all IS-IS routes. These backup routes are pre-installed in the Packet Forwarding Engine, which performs a local repair and implements the backup path when the link for a primary next hop for a particular route is no longer available. The selection of LFA is done randomly by selecting any matching LFA to progress to the given destination. This does not ensure best backup coverage available for the network. In order to choose the best LFA, Junos OS allows you to configure network-wide backup selection policies for each destination (IPv4 and IPv6) and a primary next-hop interface. These policies are evaluated based on admin-group, srlg, bandwidth, protection-type, metric, and neighbor information.

Before you begin to configure the backup selection policy for the IS-IS protocol:

- Configure the router interfaces. See the *Junos OS Network Management Administration Guide for Routing Devices*
- Configure an interior gateway protocol or static routing. See the *Junos OS Routing Protocols Library for Routing Devices*

To configure the backup selection policy for the IS-IS protocol:

1. Configure per-packet load balancing.

```
[edit policy-options]
user@host# set policy-statement ecmp term 1 then load-balance per-packet
```

2. Enable RSVP on all the interfaces.

```
[edit protocols]
user@host# set rsvp interface all
```

3. Configure administrative groups.

```
[edit protocols mpls]
user@host# set admin-groups group-name
```

4. Configure srlg values.

```
[edit routing-options]
user@host# set srlg srlg-name srlg-value srlg-value
```

5. Enable MPLS on all the interfaces.

```
[edit protocols mpls]
user@host# set interface all
```

6. Configure srlg on the interfaces.

```
[edit protocols mpls]
user@host# set interface interface-name srlg srlg-name
```

7. Configure the administrative groups on the interfaces.

```
[edit protocols mpls]
user@host# set interface interface-name admin-group group-name
```

8. Enable link protection and configure the metric value on all the interfaces.

```
[edit protocols]
user@host# set isis interface all level 2 metric 10
```

9. Apply the routing policy to all equal cost multipaths exported from the routing table to the forwarding table.

```
[edit routing-options]
user@host# set forwarding-table export ecmp
```

10. Configure the administrative group of the backup selection policy for an IP address. You can choose to exclude, include all, include any, or prefer the administrative groups from the backup path.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name admin-group
```

- Specify the administrative group to be excluded.

```
[edit routing-options backup-selection destination ip-address interface interface-name
admin-group]
user@host# set exclude group-name
```

The backup path is not selected as the loop-free alternate (LFA) or backup nexthop if any of the links in the path have any one of the listed administrative groups.

For example, to exclude the group c1 from the administrative group:

```
[edit routing-options backup-selection destination 0.0.0.0/0 interface all admin-group]
user@host# set exclude c1
```

- Configure all the administrative groups if each link in the backup path requires all the listed administrative groups in order to accept the path.

```
[edit routing-options backup-selection destination ip-address interface interface-name
admin-group]
user@host# set include-all group-name
```

For example, to set all the administrative groups if each link requires all the listed administrative groups in order to accept the path:

```
[edit routing-options backup-selection destination 0.0.0.0/0 interface all admin-group]
user@host# set include-all c2
```

- Configure any administrative group if each link in the backup path requires at least one of the listed administrative groups in order to select the path.

```
[edit routing-options backup-selection destination ip-address interface interface-name
admin-group]
user@host# set include-any group-name
```

For example, to set any administrative group if each link in the backup path requires at least one of the listed administrative groups in order to select the path:

```
[edit routing-options backup-selection destination 0.0.0.0/0 interface all admin-group]
user@host# set include-any c3
```

- Define an ordered set of administrative group that specifies the preference of the backup path.



The leftmost element in the set is given the highest preference.

```
[edit routing-options backup-selection destination ip-address interface interface-name
admin-group]
user@host# set preference group-name
```

For example, to set an ordered set of administrative group that specifies the preference of the backup path:

```
[edit routing-options backup-selection destination 0.0.0.0/0 interface all admin-group]
user@host# set preference c4
```

11. Configure the backup path to allow the selection of the backup next hop only if the bandwidth is greater than or equal to the bandwidth of the primary next hop.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name bandwidth-
greater-equal-primary
```

12. Configure the backup path to specify the metric from the one-hop neighbor or from the remote router such as an RSVP backup label-switched-path (LSP) tail-end router to the final destination. The destination metric can be either highest or lowest.

- Configure the backup path that has the highest destination metric.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name dest-
metric highest
```

- Configure the backup path that has the lowest destination metric.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name dest-
metric lowest
```

13. Configure the backup path that is a downstream path to the destination.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name downstream-
paths-only
```

14. Set the order of preference of the root and the destination metric during backup path selection. The preference order can be :

- [root dest] — Backup path selection or preference is first based on the root-metric criteria. If the criteria of all the root-metric is the same, then the selection or preference is based on the dest-metric.
- [dest root] — Backup path selection or preference is first based on the dest-metric criteria. If the criteria of all the dest-metric is the same, then the selection is based on the root-metric.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name metric-
order root
user@host# set backup-selection destination ip-address interface interface-name metric-
order dest
```

15. Configure the backup path to define a list of loop-back IP addresses of the adjacent neighbors to either exclude or prefer in the backup path selection. The neighbor can be a local (adjacent router) neighbor, remote neighbor, or any other router in the backup path.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name neighbor
```

- Configure the list of neighbors to be excluded.

```
[edit routing-options backup-selection destination ip-address interface interface-name
neighbor]
user@host# set exclude neighbor-address
```

The backup path that has a router from the list is not selected as the loop-free alternative or backup next hop.

- Configure an ordered set of neighbors to be preferred.

```
[edit routing-options backup-selection destination ip-address interface interface-name
neighbor]
user@host# set preference neighbor-address
```

The backup path having the leftmost neighbor is selected.

16. Define the backup path per-neighbor policy, to either exclude or prefer a backup path.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface all neighbor-tag
```

- Configure to not select the backup path as the loop-free alternative or backup-next hop if any node or router with route-tag is present in the path.

```
[edit routing-options backup-selection destination ip-address interface interface-name
neighbor-tag]
user@host# set exclude route-tag
```

For example, to not select the backup path as the loop-free alternative or backup-next hop if any node or router with 1004 route-tag is present in the path:

```
[edit routing-options backup-selection destination 0.0.0.0/0 interface all neighbor-tag]
user@host# set exclude 1004
```

- Configure the set of route tags in descending order of preference.

```
[edit routing-options backup-selection destination ip-address interface interface-name
neighbor-tag]
user@host# set preference route-tag
```

For example, to configure the set of route tags in descending order of preference:

```
[edit routing-options backup-selection destination 0.0.0.0/0 interface all neighbor-tag]
user@host# set preference 1006
```

17. Configure the backup path to specify the required protection type of the backup path to be link, node, or node-link.

- Select the backup path that provides link protection.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name
protection-type link
```

- Select the backup path that provides node protection.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name
protection-type node
```

- Select the backup path that allows either node or link protection LFA where node-protection LFA is preferred over link-protection LFA.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface interface-name
protection-type node-link
```

18. Specify the metric to the one-hop neighbor or to the remote router such as an RSVP backup label-switched-path (LSP) tail-end router.

- Select the path with highest root metric.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface all root-metric highest
```

- Select the path with lowest root metric.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface all root-metric lowest
```

19. Configure the backup selection path to either allow or reject the common shared risk link groups (SRLGs) between the primary link and each link in the backup path.

- Configure the backup path to allow common srlgs between the primary link and each link in the backup path.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface all srlg loose
```

A backup path with a fewer number of srlg collisions is preferred.

- Configure the backup path to reject the backup path that has common srlgs between the primary link and each link in the backup path.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface all srlg strict
```

20. Configure the backup path to control the order and the criteria of evaluating the backup path based on the administrative group, srlg, bandwidth, protection type, neighbor, neighbor-tag, and metric. The default order of evaluation is admin-group, srlg, bandwidth, protection-type, neighbor, neighbor-tag, and metric.

```
[edit routing-options]
user@host# set backup-selection destination ip-address interface all evaluation-order admin-group
user@host# set backup-selection destination ip-address interface all evaluation-order srlg
user@host# set backup-selection destination ip-address interface all evaluation-order bandwidth
```

## RELATED DOCUMENTATION

*Understanding Backup Selection Policy for IS-IS Protocol*

## Example: Redistributing OSPF Routes into IS-IS

### IN THIS SECTION

- Requirements | 152

- Overview | 152
- Configuration | 153
- Verification | 161

This example shows how to redistribute OSPF routes into an IS-IS network.

## Requirements

No special configuration beyond device initialization is required before configuring this example.

## Overview

### IN THIS SECTION

- Topology | 153

Export policy can be applied to IS-IS to facilitate route redistribution.

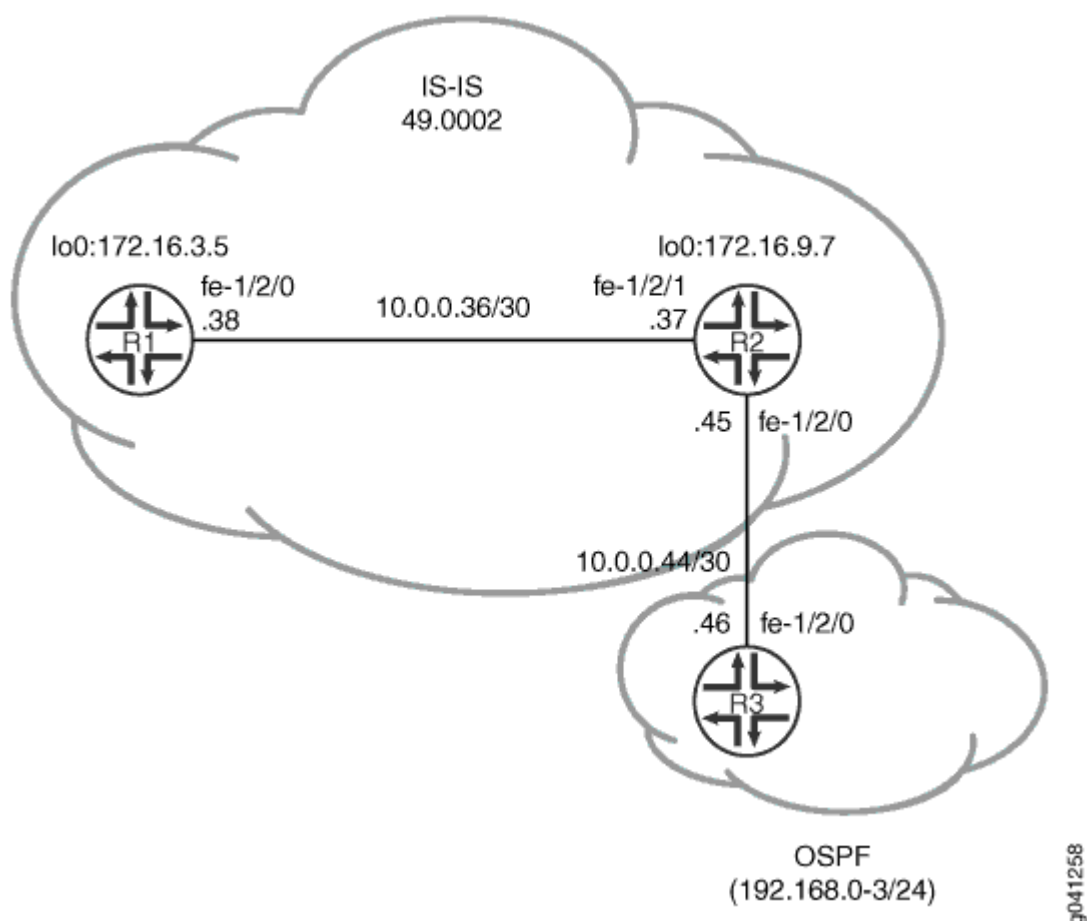
Junos OS does not support the application of import policy for link-state routing protocols like IS-IS because such policies can lead to inconsistent link-state database (LSDB) entries, which in turn can result in routing inconsistencies.

In this example, OSPF routes 192.168.0/24 through 192.168.3/24 are redistributed into IS-IS area 49.0002 from Device R2.

In addition, policies are configured to ensure that Device R1 can reach destinations on the 10.0.0.44/30 network, and that Device R3 can reach destinations on the 10.0.0.36/30 network. This enables end-to-end reachability.

[Figure 12 on page 153](#) shows the topology used in this example.

Figure 12: IS-IS Route Redistribution Topology



"CLI Quick Configuration" on page 154 shows the configuration for all of the devices in Figure 12 on page 153. The section "No Link Title" on page 155 describes the steps on Device R2. "No Link Title" on page 157 describes the steps on Device R3.

## Topology

## Configuration

### IN THIS SECTION

- Procedure | 154

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R1

```
set interfaces fe-1/2/0 unit 0 description to-R7
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.38/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 172.16.3.5/32
set interfaces lo0 unit 0 family iso address 49.0002.0172.0016.0305.00
set protocols isis interface fe-1/2/0.0
set protocols isis interface lo0.0
```

#### Device R2

```
set interfaces fe-1/2/1 unit 0 description to-R5
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.37/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/0 unit 0 description to-OSPF-network
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.45/30
set interfaces lo0 unit 0 family inet address 172.16.9.7/32
set interfaces lo0 unit 0 family iso address 49.0002.0172.0016.0907.00
set protocols isis export ospf-isis
set protocols isis export send-direct-to-isis-neighbors
set protocols isis interface fe-1/2/1.0
set protocols isis interface lo0.0
set protocols ospf export send-direct-to-ospf-neighbors
set protocols ospf area 0.0.0.1 interface fe-1/2/0.0
set protocols ospf area 0.0.0.1 interface lo0.0 passive
set policy-options policy-statement ospf-isis term 1 from protocol ospf
set policy-options policy-statement ospf-isis term 1 from route-filter 192.168.0.0/22 longer
set policy-options policy-statement ospf-isis term 1 then accept
set policy-options policy-statement send-direct-to-isis-neighbors from protocol direct
set policy-options policy-statement send-direct-to-isis-neighbors from route-filter 10.0.0.44/30
exact
set policy-options policy-statement send-direct-to-isis-neighbors then accept
set policy-options policy-statement send-direct-to-ospf-neighbors from protocol direct
```



```

set policy-options policy-statement send-direct-to-ospf-neighbors from route-filter 10.0.0.36/30
exact
set policy-options policy-statement send-direct-to-ospf-neighbors then accept

```

### Device R3

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.46/30
set interfaces lo0 unit 0 family inet address 192.168.1.1/32
set interfaces lo0 unit 0 family inet address 192.168.2.1/32
set interfaces lo0 unit 0 family inet address 192.168.3.1/32
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set protocols ospf export ospf
set protocols ospf area 0.0.0.1 interface fe-1/2/0.0
set protocols ospf area 0.0.0.1 interface lo0.0 passive
set policy-options policy-statement ospf term 1 from protocol static
set policy-options policy-statement ospf term 1 then accept
set routing-options static route 192.168.0.0/24 discard
set routing-options static route 192.168.1.0/24 discard
set routing-options static route 192.168.3.0/24 discard
set routing-options static route 192.168.2.0/24 discard

```

### Step-by-Step Procedure

To configure Device R2:

1. Configure the network interfaces.

```

[edit interfaces]
user@R2# set fe-1/2/1 unit 0 description to-R5
user@R2# set fe-1/2/1 unit 0 family inet address 10.0.0.37/30
user@R2# set fe-1/2/1 unit 0 family iso
user@R2# set fe-1/2/0 unit 0 description to-OSPF-network
user@R2# set fe-1/2/0 unit 0 family inet address 10.0.0.45/30
user@R2# set lo0 unit 0 family inet address 172.16.9.7/32
user@R2# set lo0 unit 0 family iso address 49.0002.0172.0016.0907.00

```

2. Configure IS-IS on the interface facing Device R1 and the loopback interface.

```
[edit protocols isis]
user@R2# set interface fe-1/2/1.0
user@R2# set interface lo0.0
```

3. Configure the policy that enables Device R1 to reach the 10.0.0.44/30 network.

```
[edit policy-options policy-statement send-direct-to-isis-neighbors]
user@R2# set from protocol direct
user@R2# set from route-filter 10.0.0.44/30 exact
user@R2# set then accept
```

4. Apply the policy that enables Device R1 to reach the 10.0.0.44/30 network.

```
[edit protocols isis]
user@R2# set export send-direct-to-isis-neighbors
```

5. Configure OSPF on the interfaces.

```
[edit protocols ospf]
user@R2# set area 0.0.0.1 interface fe-1/2/0.0
user@R2# set area 0.0.0.1 interface lo0.0 passive
```

6. Configure the OSPF route redistribution policy.

```
[edit policy-options policy-statement ospf-isis term 1]
user@R2# set from protocol ospf
user@R2# set from route-filter 192.168.0.0/22 longer
user@R2# set then accept
```

7. Apply the OSPF route redistribution policy to the IS-IS instance.

```
[edit protocols isis]
user@R2# set export ospf-isis
```

8. Configure the policy that enables Device R3 to reach the 10.0.0.36/30 network.

```
[edit policy-options policy-statement send-direct-to-ospf-neighbors]
user@R2# set from protocol direct
user@R2# set from route-filter 10.0.0.36/30 exact
user@R2# set then accept
```

9. Apply the policy that enables Device R3 to reach the 10.0.0.36/30 network.

```
[edit protocols ospf]
user@R2# set export send-direct-to-ospf-neighbors
```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure multi-level IS-IS:

1. Configure the network interfaces.

Multiple addresses are configured on the loopback interface to simulate multiple route destinations.

```
[edit interfaces]
user@R3# set fe-1/2/0 unit 0 family inet address 10.0.0.46/30
user@R3# set lo0 unit 0 family inet address 192.168.1.1/32
user@R3# set lo0 unit 0 family inet address 192.168.2.1/32
user@R3# set lo0 unit 0 family inet address 192.168.3.1/32
user@R3# set lo0 unit 0 family inet address 192.168.0.1/32
```

2. Configure static routes to the loopback interface addresses.

These are the routes that are redistributed into IS-IS.

```
[edit routing-options static]
user@R3# set route 192.168.0.0/24 discard
user@R3# set route 192.168.1.0/24 discard
```

```
user@R3# set route 192.168.3.0/24 discard
user@R3# set route 192.168.2.0/24 discard
```

### 3. Configure OSPF on the interfaces.

```
[edit protocols ospf area 0.0.0.1]
user@R3# set interface fe-1/2/0.0
user@R3# set interface lo0.0 passive
```

### 4. Configure the OSPF policy to export the static routes.

```
[edit policy-options policy-statement ospf term 1]
user@R3# set from protocol static
user@R3# set then accept
```

### 5. Apply the OSPF export policy.

```
[edit protocols ospf]
user@R3# set export ospf
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, `show policy-options`, and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

### Device R2

```
user@R2# show interfaces
fe-1/2/1 {
  unit 0 {
    description to-R5;
    family inet {
      address 10.0.0.37/30;
    }
    family iso;
  }
}
fe-1/2/0 {
```

```

    unit 0 {
        description to-OSPF-network;
        family inet {
            address 10.0.0.45/30;
        }
    }
}
lo0 {
    unit 0 {
        family inet {
            address 172.16.9.7/32;
        }
        family iso {
            address 49.0002.0172.0016.0907.00;
        }
    }
}
}

```

```

user@R2# show protocols
isis {
    export [ ospf-isis send-direct-to-isis-neighbors ];
    interface fe-1/2/1.0;
    interface lo0.0;
}
ospf {
    export send-direct-to-ospf-neighbors;
    area 0.0.0.1 {
        interface fe-1/2/0.0;
        interface lo0.0 {
            passive;
        }
    }
}
}

```

```

user@R2# show policy-options
policy-statement ospf-isis {
    term 1 {
        from {
            protocol ospf;
            route-filter 192.168.0.0/22 longer;

```

```

        }
        then accept;
    }
}
policy-statement send-direct-to-isis-neighbors {
    from {
        protocol direct;
        route-filter 10.0.0.44/30 exact;
    }
    then accept;
}
policy-statement send-direct-to-ospf-neighbors {
    from {
        protocol direct;
        route-filter 10.0.0.36/30 exact;
    }
    then accept;
}

```

### Device R3

```

user@R3# show interfaces
fe-1/2/0 {
    unit 0 {
        family inet {
            address 10.0.0.46/30;
        }
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.1.1/32;
            address 192.168.2.1/32;
            address 192.168.3.1/32;
            address 192.168.0.1/32;
        }
    }
}

```

```
    }
}
```

```
user@R3# show protocols
ospf {
  export ospf;
  area 0.0.0.1 {
    interface fe-1/2/0.0;
    interface lo0.0 {
      passive;
    }
  }
}
```

```
user@R3# show policy-options
policy-statement ospf {
  term 1 {
    from protocol static;
    then accept;
  }
}
```

```
user@R3# show routing-options
static {
  route 192.168.0.0/24 discard;
  route 192.168.1.0/24 discard;
  route 192.168.3.0/24 discard;
  route 192.168.2.0/24 discard;
}
```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

 [Verifying OSPF Route Advertisement | 162](#)

- [Verifying Route Redistribution | 163](#)
- [Verifying Connectivity | 164](#)

Confirm that the configuration is working properly.

## Verifying OSPF Route Advertisement

### Purpose

Make sure that the expected routes are advertised by OSPF.

### Action

From operational mode on Device R2, enter the `show route protocol ospf` command.

```
user@R2> show route protocol ospf

inet.0: 15 destinations, 15 routes (15 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.0.0/24    *[OSPF/150] 03:54:21, metric 0, tag 0
                 > to 10.0.0.46 via fe-1/2/0.0
192.168.0.1/32   *[OSPF/10] 03:54:21, metric 1
                 > to 10.0.0.46 via fe-1/2/0.0
192.168.1.0/24   *[OSPF/150] 03:54:21, metric 0, tag 0
                 > to 10.0.0.46 via fe-1/2/0.0
192.168.1.1/32   *[OSPF/10] 03:54:21, metric 1
                 > to 10.0.0.46 via fe-1/2/0.0
192.168.2.0/24   *[OSPF/150] 03:54:21, metric 0, tag 0
                 > to 10.0.0.46 via fe-1/2/0.0
192.168.2.1/32   *[OSPF/10] 03:54:21, metric 1
                 > to 10.0.0.46 via fe-1/2/0.0
192.168.3.0/24   *[OSPF/150] 03:54:21, metric 0, tag 0
                 > to 10.0.0.46 via fe-1/2/0.0
192.168.3.1/32   *[OSPF/10] 03:54:21, metric 1
                 > to 10.0.0.46 via fe-1/2/0.0
224.0.0.5/32     *[OSPF/10] 03:56:03, metric 1
```



## MultiRecv

```
iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

## Meaning

The 192.168/16 routes are advertised by OSPF.

## Verifying Route Redistribution

## Purpose

Make sure that the expected routes are redistributed from OSPF into IS-IS.

## Action

From operational mode on Device R1, enter the `show route protocol isis` command.

```
user@R1> show route protocol isis
```

```
inet.0: 13 destinations, 13 routes (13 active, 0 holddown, 0 hidden)
```

```
+ = Active Route, - = Last Active, * = Both
```

```
10.0.0.44/30      *[IS-IS/160] 03:45:24, metric 20
                  > to 10.0.0.37 via fe-1/2/0.0
172.16.9.7/32     *[IS-IS/15] 03:49:46, metric 10
                  > to 10.0.0.37 via fe-1/2/0.0
192.168.0.0/24    *[IS-IS/160] 03:49:46, metric 10
                  > to 10.0.0.37 via fe-1/2/0.0
192.168.0.1/32    *[IS-IS/160] 03:49:46, metric 11, tag2 1
                  > to 10.0.0.37 via fe-1/2/0.0
192.168.1.0/24    *[IS-IS/160] 03:49:46, metric 10
                  > to 10.0.0.37 via fe-1/2/0.0
192.168.1.1/32    *[IS-IS/160] 03:49:46, metric 11, tag2 1
                  > to 10.0.0.37 via fe-1/2/0.0
192.168.2.0/24    *[IS-IS/160] 03:49:46, metric 10
                  > to 10.0.0.37 via fe-1/2/0.0
192.168.2.1/32    *[IS-IS/160] 03:49:46, metric 11, tag2 1
                  > to 10.0.0.37 via fe-1/2/0.0
192.168.3.0/24    *[IS-IS/160] 03:49:46, metric 10
```

```

                > to 10.0.0.37 via fe-1/2/0.0
192.168.3.1/32   *[IS-IS/160] 03:49:46, metric 11, tag2 1
                > to 10.0.0.37 via fe-1/2/0.0

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

```

## Meaning

The 192.168/16 routes are redistributed into IS-IS.

## Verifying Connectivity

### Purpose

Check that Device R1 can reach the destinations on Device R3.

### Action

From operational mode, enter the ping command.

```

user@R1> ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1): 56 data bytes
64 bytes from 192.168.1.1: icmp_seq=0 ttl=63 time=2.089 ms
64 bytes from 192.168.1.1: icmp_seq=1 ttl=63 time=1.270 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=63 time=2.135 ms

```

## Meaning

These results confirm that Device R1 can reach the destinations in the OSPF network.

## Example: Configuring IS-IS Route Leaking from a Level 2 Area to a Level 1 Area

### IN THIS SECTION

- [Requirements | 165](#)
- [Overview | 165](#)
- [Configuration | 166](#)
- [Verification | 172](#)

This example shows how to leak prefixes in an IS-IS network from a Level 2 area to a Level 1 area.

### Requirements

No special configuration beyond device initialization is required before configuring this example.

### Overview

#### IN THIS SECTION

- [Topology | 165](#)

Every routing protocol passes routing information up or down the routing hierarchy. This bidirectional flow of routing information is known as route leaking.

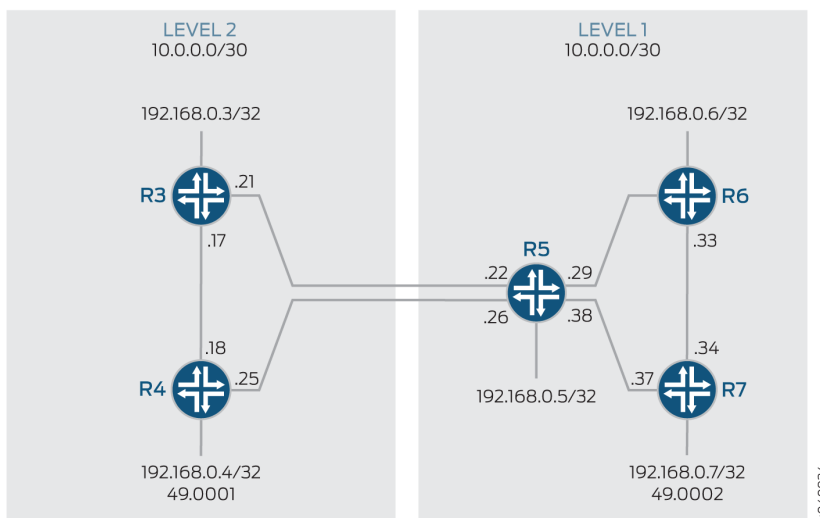
By default, IS-IS protocol leaks routing information from a Level 1 area to a Level 2 area. However, to leak routing information from a Level 2 area to a Level 1 area, an export policy must be explicitly configured.

### Topology

In this example, Devices R3 and R4 are configured in a Level 2 area. Devices R5, R6, and R7 are configured in a Level 1 area.

[Figure 13 on page 166](#) shows the topology used in this example.

Figure 13: Route Leaking from a Level 2 Area to a Level 1 Area



## Configuration

### IN THIS SECTION

- [Configuring Route Leaking from a Level 2 Area to a Level 1 Area | 166](#)
- [Configuring Route Leaking from a Level 2 Area to a Level 1 Area | 169](#)
- [Results | 170](#)

## Configuring Route Leaking from a Level 2 Area to a Level 1 Area

### CLI Quick Configuration

To quickly configure route leaking from a Level 2 area to a Level 1 area, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R3

```
set interfaces fe-1/2/0 unit 0 description to-R4
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.17/30
set interfaces fe-1/2/0 unit 0 family iso
```

```

set interfaces fe-1/2/1 unit 0 description to-R5
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.21/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0001.0192.0168.0003.00
set policy-options policy-statement leak-L2-to-L1 from route-filter 192.168.0.0/24 orlonger
set policy-options policy-statement leak-L2-to-L1 from protocol isis
set policy-options policy-statement leak-L2-to-L1 from level 2
set policy-options policy-statement leak-L2-to-L1 to protocol isis
set policy-options policy-statement leak-L2-to-L1 to level 1
set policy-options policy-statement leak-L2-to-L1 then accept
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0 level 1 disable
set protocols isis export leak-L2-to-L1

```

#### Device R4

```

set interfaces fe-1/2/0 unit 0 description to-R3
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.18/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R5
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.25/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.4/32
set interfaces lo0 unit 0 family iso address 49.0001.0192.0168.0004.00
set policy-options policy-statement leak-L2-to-L1 from route-filter 192.168.0.0/24 orlonger
set policy-options policy-statement leak-L2-to-L1 from protocol isis
set policy-options policy-statement leak-L2-to-L1 from level 2
set policy-options policy-statement leak-L2-to-L1 to protocol isis
set policy-options policy-statement leak-L2-to-L1 to level 1
set policy-options policy-statement leak-L2-to-L1 then accept
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0 level 1 disable
set protocols isis export leak-L2-to-L1

```

#### Device R5

```

set interfaces fe-1/2/0 unit 0 description to-R3
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.22/30

```

```

set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R4
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.26/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/2 unit 0 description to-R6
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.29/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces fe-1/2/3 unit 0 description to-R7
set interfaces fe-1/2/3 unit 0 family inet address 10.0.0.38/30
set interfaces fe-1/2/3 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.5/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0005.00
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface fe-1/2/2.0 level 2 disable
set protocols isis interface fe-1/2/3.0 level 2 disable
set protocols isis interface lo0.0 level 1 disable

```

## Device R6

```

set interfaces fe-1/2/0 unit 0 description to-R5
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.30/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R7
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.33/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.6/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0006.00
set protocols isis interface fe-1/2/0.0 level 2 disable
set protocols isis interface fe-1/2/1.0 level 2 disable
set protocols isis interface lo0.0 level 2 disable

```

## Device R7

```

set interfaces fe-1/2/0 unit 0 description to-R6
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.34/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R5
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.37/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.7/32

```

```

set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0007.00
set protocols isis interface fe-1/2/0.0 level 2 disable
set protocols isis interface fe-1/2/1.0 level 2 disable
set protocols isis interface lo0.0 level 2 disable

```

## Step-by-Step Procedure

### Configuring Route Leaking from a Level 2 Area to a Level 1 Area

## Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [Junos OS CLI User Guide](#).

To configure route leaking from a Level 2 area to a Level 1 area:

#### 1. Configure the network interfaces.

Enable IS-IS on the interfaces by including the ISO address family on each interface.

```

[edit interfaces]
user@R3# set fe-1/2/0 unit 0 description to-R4
user@R3# set fe-1/2/0 unit 0 family inet address 10.0.0.17/30
user@R3# set fe-1/2/0 unit 0 family iso
user@R3# set fe-1/2/1 unit 0 description to-R5
user@R3# set fe-1/2/1 unit 0 family inet address 10.0.0.21/30
user@R3# set fe-1/2/1 unit 0 family iso

```

Similarly, configure other routers.

#### 2. Configure two loopback interface addresses.

One address is for IPv4, and the other address is to enable the router to form adjacencies with other routers in the area.

```

[edit interfaces lo0 unit 0]
user@R3# set family inet address 192.168.0.3/32
user@R3# set family iso address 49.0001.0192.0168.0003.00

```

### 3. Specify the IS-IS level on a per-interface basis.

```
[edit protocols isis interface]
user@R3# set fe-1/2/0.0 level 1 disable
user@R3# set fe-1/2/1.0 level 1 disable
user@R3# set lo0.0 level 1 disable
```

### 4. Configure a route leaking policy on the routers configured in the Level 2 area to leak routes into the Level 1 area.

```
[edit policy-options policy-statement leak-L2-to-L1]
user@R3# set from route-filter 192.168.0.0/24 orlonger
user@R3# set from protocol isis
user@R3# set from level 2
user@R3# set to protocol isis
user@R3# set to level 1
user@R3# set then accept
```

```
[edit protocols isis]
user@R3# set export leak-L2-to-L1
```

Similarly, configure Device R4.

## Results

From configuration mode, confirm your configuration by entering the **show interfaces**, **show protocols isis**, and **show policy-options** commands.

If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R3# show interfaces
fe-1/2/0 {
    unit 0 {
        description to-R4;
        family inet {
            address 10.0.0.17/30;
        }
        family iso;
```



```

    }
}
fe-1/2/1 {
    unit 0 {
        description to-R5;
        family inet {
            address 10.0.0.21/30;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.0.3/32;
        }
        family iso {
            address 49.0001.0192.0168.0003.00;
        }
    }
}
}

```

```

user@R3# show protocols isis
export leak-L2-to-L1;
interface fe-1/2/0.0 {
    level 1 disable;
}
interface fe-1/2/1.0 {
    level 1 disable;
}
interface lo0.0 {
    level 1 disable;
}

```

```

user@R3# show policy-options
policy-statement leak-L2-to-L1 {
    from {
        protocol isis;
        level 2;
    }
}

```

```
        route-filter 192.168.0.0/24 orlonger;
    }
    to {
        protocol isis;
        level 1;
    }
    then accept;
}
```

Similarly, confirm the configuration on all other routers. If you are done configuring the routers, enter `commit` from configuration mode.

### Verification

#### IN THIS SECTION

- [Verifying Route Leaking from a Level 2 Area to a Level 1 Area | 172](#)

### Verifying Route Leaking from a Level 2 Area to a Level 1 Area

#### Purpose

Verify that IS-IS leaks routes from a Level 2 area to a Level 1 area.

#### Action

To verify that route leaking is taking place, use the following commands:

- **show isis adjacency** (to verify that the IS-IS network is up and adjacencies have been established)
- **show isis database detail** (to verify the presence of leaked routes)

1. From operational mode on Device R3, run the **show isis adjacency** command.

```
user@R3> show isis adjacency
```

Interface	System	L State	Hold (secs)	SNPA
fe-1/2/0.0	R4	2 Up	7	0:5:85:8f:94:bd
fe-1/2/1.0	R5	2 Up	7	0:5:85:8f:94:bd

The output verifies that the interfaces on Device R3 are up and have established adjacencies with the connecting interfaces on Routers R4 and R5. If you don't see the interfaces being functional, see the ["Results" on page 170](#) section for troubleshooting your configuration.

2. From operational mode on Device R3, run the **show isis database detail** command.

```
user@R3> show isis database detail
IS-IS level 1 link-state database:

R3.00-00 Sequence: 0x19, Checksum: 0x3453, Lifetime: 1078 secs
  IP prefix: 192.168.0.4/32          Metric:      10 Internal Down
  IP prefix: 192.168.0.5/32          Metric:      10 Internal Down
  IP prefix: 192.168.0.6/32          Metric:      20 Internal Down
  IP prefix: 192.168.0.7/32          Metric:      20 Internal Down

IS-IS level 2 link-state database:

R3.00-00 Sequence: 0x1c, Checksum: 0xc657, Lifetime: 1078 secs
  IS neighbor: R4.02                 Metric:      10
  IS neighbor: R5.02                 Metric:      10
  IP prefix: 10.0.0.16/30             Metric:      10 Internal Up
  IP prefix: 10.0.0.20/30             Metric:      10 Internal Up
  IP prefix: 192.168.0.3/32           Metric:      0 Internal Up

R4.00-00 Sequence: 0x19, Checksum: 0xea13, Lifetime: 1076 secs
  IS neighbor: R4.02                 Metric:      10
  IS neighbor: R5.03                 Metric:      10
  IP prefix: 10.0.0.16/30             Metric:      10 Internal Up
  IP prefix: 10.0.0.24/30             Metric:      10 Internal Up
  IP prefix: 192.168.0.4/32           Metric:      0 Internal Up

R4.02-00 Sequence: 0x17, Checksum: 0xecab, Lifetime: 1076 secs
  IS neighbor: R3.00                 Metric:      0
  IS neighbor: R4.00                 Metric:      0

R5.00-00 Sequence: 0x12, Checksum: 0xf4e5, Lifetime: 1076 secs
  IS neighbor: R5.02                 Metric:      10
  IS neighbor: R5.03                 Metric:      10
  IP prefix: 10.0.0.20/30             Metric:      10 Internal Up
  IP prefix: 10.0.0.24/30             Metric:      10 Internal Up
  IP prefix: 10.0.0.28/30             Metric:      10 Internal Up
  IP prefix: 10.0.0.32/30             Metric:      20 Internal Up
```

IP prefix: 10.0.0.36/30	Metric: 10 Internal Up
IP prefix: 192.168.0.5/32	Metric: 0 Internal Up
IP prefix: 192.168.0.6/32	Metric: 10 Internal Up
IP prefix: 192.168.0.7/32	Metric: 10 Internal Up

R5.02-00 Sequence: 0xb, Checksum: 0x2d74, Lifetime: 1076 secs

IS neighbor: R3.00	Metric: 0
--------------------	-----------

IS neighbor: R5.00	Metric: 0
--------------------	-----------

R5.03-00 Sequence: 0xb, Checksum: 0x6c32, Lifetime: 1076 secs

IS neighbor: R4.00	Metric: 0
--------------------	-----------

IS neighbor: R5.00	Metric: 0
--------------------	-----------

The **Down** keyword identifies the routes that have successfully leaked from the Level 2 area to the Level 1 area.

## Meaning

Route leaking from a Level 2 to a Level 1 area is functioning as expected.

## Handling of the IS-IS Binding SID S Flag and RFC 7794 Prefix Attribute Flags

### IN THIS SECTION

- [Benefits of IS-IS binding SID S flag and RFC 7794 prefix attribute flags: | 175](#)

As part of the SR support, the SID/Label Binding TLV originates on any router in an IS-IS domain. There are multiple uses of the SID/Label Binding TLV. One use case is that the SID/Label Binding TLV could be used to advertise prefixes to SID/Label mappings. This functionality is called the Segment Routing Mapping Server (SRMS). LDP mapping client/server is the feature that enables LDP and SR, interoperate in a network with islands of LDP and SPRING capable routers.

Any router in the domain could be configured as SRMS server. To leak binding SIDs across levels you need to configure one node as SRMS server in each IS-IS level. If the S flag is set, then the SRMS binding SIDs are leaked across the levels by the ABR (Area Border Router).

IS-IS binding SIDs supports the following in IS-IS SRMS:

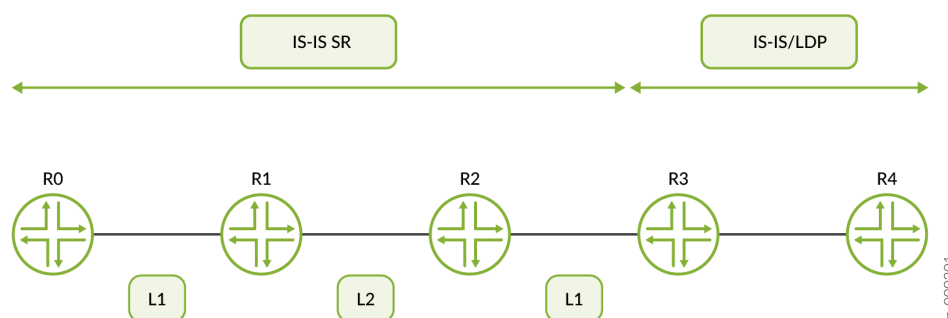
- To leak binding SIDs across IS-IS levels if the S flag is set.
- Setting and resetting of A, D flags when binding SIDs are leaked across levels.
- Support prefix attribute flags sub TLV of RFC 7794.

### Benefits of IS-IS binding SID S flag and RFC 7794 prefix attribute flags:

- Advertise binding SIDs in single node in the entire domain, rather than configuring and advertising at each level.
- Allows you to disable the binding SIDs leakage across IS-IS levels.
- Ability to control the binding SIDs leaking and determine the traffic route with labels/prefixes by setting the A and D flags.
- Set the prefix attribute flags on the receiving router to have accurate information on the routes.

The IS-IS leaks the binding SIDs by default if the S flag is set. You can disable IS-IS binding SIDs leak by including the `no-binding-sid-leaking` configuration statement at the [edit protocols isis source-packet-routing] hierarchy level. To configure and support the encoding of the prefix attribute flags sub tlv include the `prefix-attribute-flags` configuration statement at the [edit policy-options policy-statement <policy-name> then] hierarchy level. You can use the options [node-segment-flag | external-prefix-flag] for the `prefix-attribute-flags` configuration statement.

**Figure 14: IS-IS binding SID 'S' flag and RFC 7794 prefix attribute flags**



In this topology, R0, R1, and R2 are configured with IS-IS Segment Routing protocol. R3 is configured with IS-IS, SR and LDP. R4 is configured with IS-IS and LDP, but without SR. R1 and R2 are Area Border Routers (ABRs).

R0 participates at Level-1 of IS-IS domain. R1 and R2 participate with IS-IS SR at Level 1 and Level 2. R3 participates in Level 1 and advertises the binding SIDs by acting as a SRMS server. R2 leaks the binding SIDs to level 2. R1 leaks the binding SIDs to level 1.

You can set the S flag to allow the label binding TLV to leak through the IS-IS levels (Level 1 or Level 2). You can set the A flag to program the penultimate-hop popping. During leaking binding SID, the D flag is set to prevent the leaking of the label binding TLV from Level 2 back to Level 1.

## Understanding BGP Communities, Extended Communities, and Large Communities as Routing Policy Match Conditions

A *BGP community* is a group of destinations that share a common property. Community information is included as a path attribute in BGP update messages. This information identifies community members and enables you to perform actions on a group without having to elaborate upon each member. You can use community and extended communities attributes to trigger routing decisions, such as acceptance, rejection, preference, or redistribution.

You can assign community tags to non-BGP routes through configuration (for static, aggregate, or generated routes) or an import routing policy. These tags can then be matched when BGP exports the routes.

A community value is a 32-bit field that is divided into two main sections. The first 16 bits of the value encode the AS number of the network that originated the community, while the last 16 bits carry a unique number assigned by the AS. This system attempts to guarantee a globally unique set of community values for each AS in the Internet. Junos OS uses a notation of *as-number:community-value*, where each value is a decimal number. The AS values of 0 and 65,535 are reserved, as are all of the community values within those AS numbers. Each community, or set of communities, is given a name within the [edit policy-options] configuration hierarchy. The name of the community uniquely identifies it to the routing device and serves as the method by which routes are categorized. For example, a route with a community value of 64510:1111 might belong to the community named AS64510-routes. The community name is also used within a routing policy as a match criterion or as an action. The command syntax for creating a community is: `policy-options community name members [community-ids]`. The *community-ids* are either a single community value or multiple community values. When more than one value is assigned to a community name, the routing device interprets this as a logical AND of the community values. In other words, a route must have all of the configured values before being assigned the community name.

The regular community attribute is four octets. Networking enhancements, such as VPNs, have functionality requirements that can be satisfied by an attribute such as a community. However, the 4-octet community value does not provide enough expansion and flexibility to accommodate VPN requirements. This leads to the creation of extended communities. An extended community is an 8-octet value that is also divided into two main sections. The first 2 octets of the community encode a

type field while the last 6 octets carry a unique set of data in a format defined by the type field. Extended communities provide a larger range for grouping or categorizing communities.

The BGP extended communities attribute format has three fields: *type:administrator:assigned-number*. The routing device expects you to use the words *target* or *origin* to represent the type field. The administrator field uses a decimal number for the AS or an IPv4 address, while the assigned number field expects a decimal number no larger than the size of the field (65,535 for 2 octets or 4,294,967,295 for 4 octets).

When specifying community IDs for standard and extended community attributes, you can use UNIX-style regular expressions. The only exception is for VPN import policies (*vrf-import*), which do not support regular expressions for the extended communities attribute.

Regular BGP communities attributes are a variable length attribute consisting of a set of one or more 4-byte values that was split into 16 bit values. The most significant word is interpreted as an AS number and least significant word is a locally defined value assigned by the operator of the AS. Since the adoption of 4-byte ASNs, the 4-byte BGP regular community and 6-byte BGP extended community can no longer support BGP community attributes. Operators often encode AS number in the local portion of the BGP community that means that sometimes the format of the community is ASN:ASN. With the 4-byte ASN, you need 8-bytes to encode it. Although BGP extended community permits a 4-byte AS to be encoded as the global administrator field, the local administrator field has only 2-byte of available space. Thus, 6-byte extended community attribute is also unsuitable. To overcome this, Junos OS allows you to configure optional transitive path attribute - a 12-byte BGP large community that provides the most significant 4-byte value to encode autonomous system number as the global administrator and the remaining two 4-byte assigned numbers to encode the local values as defined in RFC 8092. You can configure BGP large community at the [edit policy-options community *community-name* members] and [edit routing-options static route *ip-address* community] hierarchy levels. The BGP large community attributes format has four fields: *large:global administrator:assigned number:assigned number*.

The BGP IPv6 unicast address specific extended community are encoded as a set of 20-bytes value. The 20-byte value gets interpreted in the following format:

- Most significant 2-bytes encodes the Type and Sub-Type value (high value (most significant byte) and Low value (second most significant byte)).
- Next 16-bytes encodes the IPv6 unicast address. It is the global administrator in the IETF RFC.
- Last 2-bytes encodes the operator defined local values. It is local administrator in the IETF RFC.

The IPv6 unicast address specific BGP extended community attributes are represented by a keyword *ipv6-target*, *ipv6-origin*, or *ipv6-extended* followed by IPv6 and local administrator separated by *<*, *>*, and *..*.



**NOTE:** The length of the BGP large communities attribute value should be a non-zero multiple of 12.

## RELATED DOCUMENTATION

*Understanding How to Define BGP Communities and Extended Communities*

*How BGP Communities and Extended Communities Are Evaluated in Routing Policy Match Conditions*

*Example: Configuring a Routing Policy That Removes BGP Communities*

*Example: Configuring Communities in a Routing Policy*

*Example: Configuring Extended Communities in a Routing Policy*

## Example: Configuring a Routing Policy to Redistribute BGP Routes with a Specific Community Tag into IS-IS

### IN THIS SECTION

- [Requirements | 178](#)
- [Overview | 178](#)
- [Configuration | 179](#)
- [Verification | 190](#)

This example defines a policy that takes BGP routes from the Edu community and places them into IS-IS with a metric of 63.

### Requirements

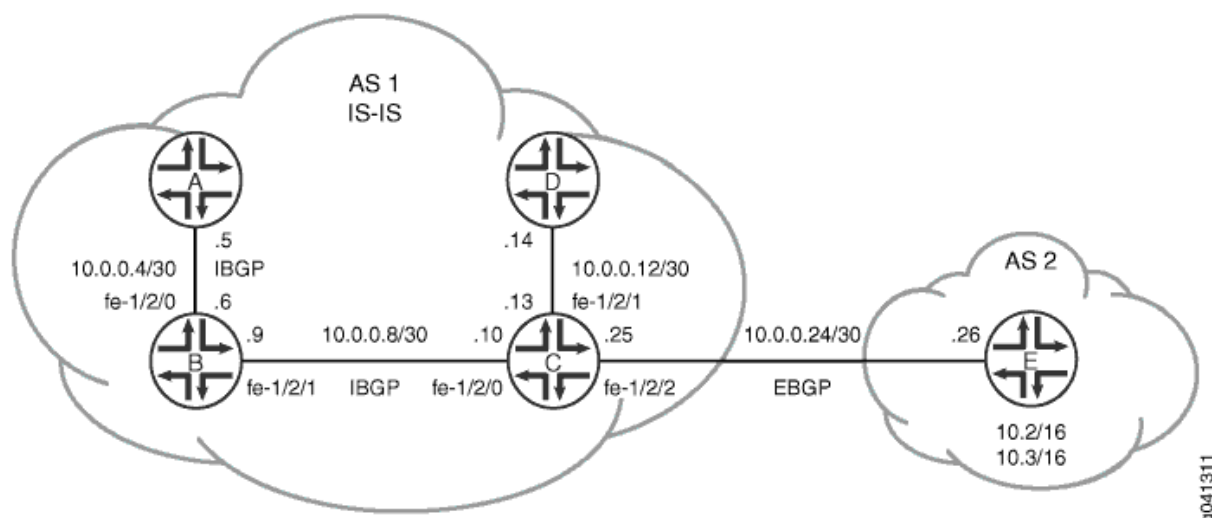
No special configuration beyond device initialization is required before configuring this example.

### Overview

[Figure 15 on page 179](#) shows the topology used in this example.



**Figure 15: Redistributing BGP Routes with a Specific Community Tag into IS-IS**



In this example, Device A, Device B, Device C, and Device D are in autonomous system (AS) 1 and are running IS-IS. All of the AS 1 devices, except Device D, are running internal BGP (IBGP).

Device E is in AS 2 and has an external BGP (EBGP) peering session with Device C. Device E has two static routes, 10.2.0.0/16 and 10.3.0.0/16. These routes are tagged with the Edu 2:5 community attribute and are advertised by way of EBGP to Device C.

Device C accepts the BGP routes that are tagged with the Edu 2:5 community attribute, redistributes the routes into IS-IS, and applies an IS-IS metric of 63 to these routes.

"CLI Quick Configuration" on page 180 shows the configuration for all of the devices in Figure 15 on page 179. The section "No Link Title" on page 183 describes the steps on Device C and Device E.

## Configuration

### IN THIS SECTION

- Procedure | 180

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device A

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.5/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.1
set protocols bgp group int neighbor 192.168.0.2
set protocols bgp group int neighbor 192.168.0.3
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.1
set routing-options autonomous-system 1
```

#### Device B

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.6/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.9/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0002.00
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.2
set protocols bgp group int neighbor 192.168.0.1
set protocols bgp group int neighbor 192.168.0.3
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.2
set routing-options autonomous-system 1
```

## Device C

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.10/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.13/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.25/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0003.00
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.3
set protocols bgp group int neighbor 192.168.0.1
set protocols bgp group int neighbor 192.168.0.2
set protocols bgp group external-peers type external
set protocols bgp group external-peers export send-isis-and-direct
set protocols bgp group external-peers peer-as 2
set protocols bgp group external-peers neighbor 10.0.0.26
set protocols isis export Edu-to-isis
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface fe-1/2/2.0 level 1 disable
set protocols isis interface fe-1/2/2.0 level 2 passive
set protocols isis interface lo0.0
set policy-options policy-statement Edu-to-isis term 1 from protocol bgp
set policy-options policy-statement Edu-to-isis term 1 from community Edu
set policy-options policy-statement Edu-to-isis term 1 then metric 63
set policy-options policy-statement Edu-to-isis term 1 then accept
set policy-options policy-statement send-isis-and-direct term 1 from protocol isis
set policy-options policy-statement send-isis-and-direct term 1 from protocol direct
set policy-options policy-statement send-isis-and-direct term 1 from route-filter 10.0.0.0/16
orlonger
set policy-options policy-statement send-isis-and-direct term 1 from route-filter 192.168.0.0/16
orlonger
set policy-options policy-statement send-isis-and-direct term 1 then accept
set policy-options community Edu members 2:5
set routing-options router-id 192.168.0.3
set routing-options autonomous-system 1

```

## Device D

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.14/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.4/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0004.00
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.4
set routing-options autonomous-system 1
```

## Device E

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.26/30
set interfaces lo0 unit 7 family inet address 192.168.0.5/32 primary
set interfaces lo0 unit 7 family inet address 10.2.0.1/32
set interfaces lo0 unit 7 family inet address 10.3.0.1/32
set protocols bgp group external-peers type external
set protocols bgp group external-peers export statics
set protocols bgp group external-peers peer-as 1
set protocols bgp group external-peers neighbor 10.0.0.25
set policy-options policy-statement statics from protocol static
set policy-options policy-statement statics then community add Edu
set policy-options policy-statement statics then accept
set policy-options community Edu members 2:5
set routing-options static route 10.2.0.0/16 reject
set routing-options static route 10.2.0.0/16 install
set routing-options static route 10.3.0.0/16 reject
set routing-options static route 10.3.0.0/16 install
set routing-options router-id 192.168.0.5
set routing-options autonomous-system 2
```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure Device E:

1. Configure the interfaces.

```
[edit interfaces]
user@E# set fe-1/2/0 unit 0 family inet address 10.0.0.26/30
user@E# set lo0 unit 7 family inet address 192.168.0.5/32 primary
user@E# set lo0 unit 7 family inet address 10.2.0.1/32
user@E# set lo0 unit 7 family inet address 10.3.0.1/32
```

2. Configure the statics policy, which adds the Edu community attribute to the static routes.

```
[edit policy-options]
user@E# set policy-statement statics from protocol static
user@E# set policy-statement statics then community add Edu
user@E# set policy-statement statics then accept
user@E# set community Edu members 2:5
```

3. Configure EBGp and apply the statics policy.

```
[edit protocols bgp group external-peers]
user@E# set type external
user@E# set export statics
user@E# set peer-as 1
user@E# set protocols bgp group external-peers neighbor 10.0.0.25
```

4. Configure the static routes.

```
[edit routing-options static]
user@E# set route 10.2.0.0/16 reject
user@E# set route 10.2.0.0/16 install
user@E# set route 10.3.0.0/16 reject
user@E# set route 10.3.0.0/16 install
```

5. Configure the router ID and the AS number.

```
[edit routing-options]
user@E# set router-id 192.168.0.5
user@E# set autonomous-system 2
```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure Device C:

### 1. Configure the interfaces.

```
[edit interfaces]
user@C# set fe-1/2/0 unit 0 family inet address 10.0.0.10/30
user@C# set fe-1/2/0 unit 0 family iso
user@C# set fe-1/2/1 unit 0 family inet address 10.0.0.13/30
user@C# set fe-1/2/1 unit 0 family iso
user@C# set fe-1/2/2 unit 0 family inet address 10.0.0.25/30
user@C# set fe-1/2/2 unit 0 family iso
user@C# set lo0 unit 0 family inet address 192.168.0.3/32
user@C# set lo0 unit 0 family iso address 49.0002.0192.0168.0003.00
```

### 2. Configure IBGP.

```
[edit protocols bgp group int]
user@C# set type internal
user@C# set local-address 192.168.0.3
user@C# set neighbor 192.168.0.1
user@C# set neighbor 192.168.0.2
```

### 3. Configure the Edu-to-isis policy, which redistributes the Edu-tagged BGP routes learned from Device E and applies a metric of 63.

```
[edit policy-options]
user@C# set policy-statement Edu-to-isis term 1 from protocol bgp
user@C# set policy-statement Edu-to-isis term 1 from community Edu
user@C# set policy-statement Edu-to-isis term 1 then metric 63
user@C# set policy-statement Edu-to-isis term 1 then accept
user@C# set community Edu members 2:5
```

4. Enable IS-IS on the interfaces, and apply the Edu-to-isis policy.

```
[edit protocols isis]
user@C# set export Edu-to-isis
user@C# set interface fe-1/2/0.0 level 1 disable
user@C# set interface fe-1/2/1.0 level 1 disable
user@C# set interface fe-1/2/2.0 level 1 disable
user@C# set interface fe-1/2/2.0 level 2 passive
user@C# set interface lo0.0
```

5. Configure the send-isis-and-direct policy, which redistributes routes to Device E, through EBGp.

Without this policy, Device E would not have connectivity to the networks in AS 1.

```
[edit policy-options policy-statement send-isis-and-direct term 1]
user@C# set from protocol isis
user@C# set from protocol direct
user@C# set from route-filter 10.0.0.0/16 orlonger
user@C# set from route-filter 192.168.0.0/16 orlonger
user@C# set then accept
```

6. Configure EBGp and apply the send-isis-and-direct policy.

```
[edit protocols bgp group external-peers]
user@C# set type external
user@C# set export send-isis-and-direct
user@C# set peer-as 2
user@C# set neighbor 10.0.0.26
```

7. Configure the router ID and the autonomous system (AS) number.

```
[edit routing-options]
user@C# set router-id 192.168.0.3
user@C# set autonomous-system 1
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, `show policy-options`, and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

### Device E

```
user@E# show interfaces
fe-1/2/0 {
  unit 0 {
    family inet {
      address 10.0.0.26/30;
    }
  }
}
lo0 {
  unit 0 {
    family inet {
      address 192.168.0.5/32 {
        primary;
      }
      address 10.2.0.1/32;
      address 10.3.0.1/32;
    }
  }
}
```

```
user@E# show protocols
bgp {
  group external-peers {
    type external;
    export statics;
    peer-as 1;
    neighbor 10.0.0.25;
  }
}
```

```
user@E# show policy-options
policy-statement statics {
```



```

    from protocol static;
    then {
        community add Edu;
        accept;
    }
}
community Edu members 2:5;

```

```

user@E# show routing-options
static {
    route 10.2.0.0/16 {
        reject;
        install;
    }
    route 10.3.0.0/16 {
        reject;
        install;
    }
}
router-id 192.168.0.5;
autonomous-system 2;

```

## Device C

```

user@C# show interfaces
fe-1/2/0 {
    unit 0 {
        family inet {
            address 10.0.0.10/30;
        }
        family iso;
    }
}
fe-1/2/1 {
    unit 0 {
        family inet {
            address 10.0.0.13/30;
        }
        family iso;
    }
}

```

```

fe-1/2/2 {
    unit 0 {
        family inet {
            address 10.0.0.25/30;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.0.3/32;
        }
        family iso {
            address 49.0002.0192.0168.0003.00;
        }
    }
}

```

```

user@C# show protocols
bgp {
    group int {
        type internal;
        local-address 192.168.0.3;
        neighbor 192.168.0.1;
        neighbor 192.168.0.2;
    }
    group external-peers {
        type external;
        export send-isis-and-direct;
        peer-as 2;
        neighbor 10.0.0.26;
    }
}
isis {
    export Edu-to-isis;
    interface fe-1/2/0.0 {
        level 1 disable;
    }
    interface fe-1/2/1.0 {
        level 1 disable;
    }
}

```

```

    }
    interface fe-1/2/2.0 {
        level 1 disable;
        level 2 passive;
    }
    interface lo0.0;
}

```

```

user@C# show policy-options
policy-statement Edu-to-isis {
    term 1 {
        from {
            protocol bgp;
            community Edu;
        }
        then {
            metric 63;
            accept;
        }
    }
}
policy-statement send-isis-and-direct {
    term 1 {
        from {
            protocol [ isis direct ];
            route-filter 10.0.0.0/16 orlonger;
            route-filter 192.168.0.0/16 orlonger;
        }
        then accept;
    }
}
community Edu members 2:5;

```

```

user@C# show routing-options
router-id 192.168.0.3;
autonomous-system 1;

```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Verifying the IS-IS Neighbor | 190](#)

Confirm that the configuration is working properly.

### Verifying the IS-IS Neighbor

#### Purpose

Verify that the BGP routes from Device E are communicated on the IS-IS network in AS 1.

#### Action

From operational mode, enter the `show route protocol isis` command.

```
user@D> show route protocol isis
inet.0: 11 destinations, 11 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.4/30      *[IS-IS/18] 22:30:53, metric 30
                 > to 10.0.0.13 via fe-1/2/0.0
10.0.0.8/30      *[IS-IS/18] 22:30:53, metric 20
                 > to 10.0.0.13 via fe-1/2/0.0
10.0.0.24/30     *[IS-IS/18] 03:31:21, metric 20
                 > to 10.0.0.13 via fe-1/2/0.0
10.2.0.0/16      *[IS-IS/165] 02:36:31, metric 73
                 > to 10.0.0.13 via fe-1/2/0.0
10.3.0.0/16      *[IS-IS/165] 02:36:31, metric 73
                 > to 10.0.0.13 via fe-1/2/0.0
192.168.0.1/32   *[IS-IS/18] 03:40:28, metric 30
                 > to 10.0.0.13 via fe-1/2/0.0
192.168.0.2/32   *[IS-IS/18] 22:30:53, metric 20
                 > to 10.0.0.13 via fe-1/2/0.0
192.168.0.3/32   *[IS-IS/18] 22:30:53, metric 10
                 > to 10.0.0.13 via fe-1/2/0.0
```

```
iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

## Meaning

As expected, the 10.2.0.0/16 and 10.3.0.0/16 routes are in Device D's routing table as IS-IS external routes with a metric of 73. If Device C had not added 63 to the metric, Device D would have a metric of 10 for these routes.

## RELATED DOCUMENTATION

[Advertising LSPs into IGP](#)

## IS-IS Extensions to Support Route Tagging

To control the transmission of routes into IS-IS, or to control transmission of IS-IS routes between different IS-IS levels, you can tag routes with certain attributes. IS-IS routes can carry these attributes, which the routing policies can use to export and import routes between different IS-IS levels. A sub-TLV to the IP prefix TLV is used to carry the tag or attribute on the routes.



**NOTE:** Route tagging does not work when IS-IS traffic engineering is disabled.

```
protocols {
  isis {
    export tag-lo0;
  }
}
policy-options {
  policy-statement tag-lo0 {
    from {
      interface lo0.0;
    }
    then {
      accept;
      tag 200;
    }
  }
}
```

```
    }
}
```

You can verify that the tag has been correctly applied by using the `show isis database extensive` command. In the command output, look for the `Administrative tag` field.

After verifying that the routes are tagged correctly, you can apply a route leaking policy to match against the presence of administrative tags, rather than specifying a list of route filters.

```
protocols {
    isis {
        export leak-tagged-L2-to-L1;
    }
}
policy-options {
    policy-statement leak-tagged-L2-to-L1 {
        from {
            tag 200;
            protocol isis;
            level 2;
        }
        to {
            protocol isis;
            level 1;
        }
        then accept;
    }
}
```

## RELATED DOCUMENTATION

[Example: Configuring IS-IS Route Leaking from a Level 2 Area to a Level 1 Area](#) | 165

## Example: Configuring a Routing Policy to Prioritize IS-IS Routes

### IN THIS SECTION

- [Requirements | 193](#)
- [Overview | 193](#)
- [Configuration | 195](#)
- [Verification | 202](#)

In a network with a large number of IS-IS routes, it can be useful to control the order in which routes are updated in response to a network topology change. This example shows how to define a routing policy to prioritize some IS-IS routes over others. In the event of an IS-IS topology change, high priority prefixes are updated in the routing table first, followed by medium and then low priority prefixes. Internet Service Providers (ISP) can use this feature to ensure faster convergence for important customers.

### Requirements

No special configuration beyond device initialization is required before configuring this example.

This example uses the following hardware and software components:

- Three routers
- Junos OS running on all devices

### Overview

#### IN THIS SECTION

- [Topology | 194](#)

You can prioritize or reject IS-IS routes that are installed in the routing table. Use the reject policy option to reject routes from a specific prefix or routes marked with a particular tag.

You can prioritize IS-IS routes for better convergence and to provide differentiated services. In a network with a large number of IGP prefixes with BGP Layer 3 VPN or label-based pseudowire service

established on top of some IGP prefixes, it is important to control the order in which routes get updated in the forwarding table. You can configure an import policy and use a route tag or filter the routes based on their prefix before setting a priority of high, medium, or low as per your network requirements. The IS-IS protocol downloads routes to the rpd routing table based on the configured priority. If you do not configure an import policy, all routes are set to a medium priority by default.

An IS-IS import policy can be used to set priority or to filter IS-IS external routes based on the following criteria:

**Prefix** Use `route-filter` policy option to filter known prefixes.

**Route Tag** Use `tag` policy option to assign a specific priority for prefixes that contain a particular tag.



**NOTE:** If an IS-IS import policy is applied that results in a reject terminating action for a non-external route, then the reject action is ignored and the route is accepted anyway. By default, such a route is now installed in the routing table with a priority of low. This behavior prevents traffic black holes, that is, silently discarded traffic, by ensuring consistent routing. However, you can use the `reject` policy option to reject routes based on the prefix or the configured tag.



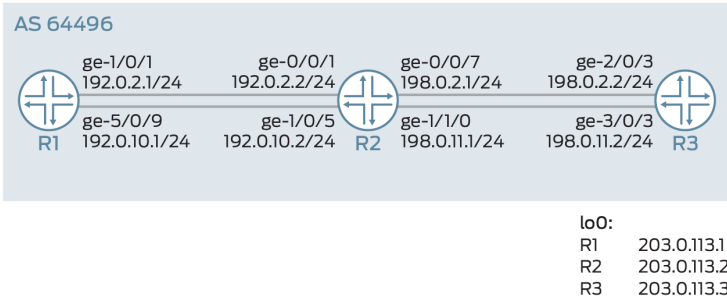
**CAUTION:** You might see an increase in micro loop traffic as order of route download changes.

## Topology

In [Figure 16 on page 195](#), Router R1 is connected to Router R3 via Router R2. We need to set a high priority to a route to Router R3 to ensure quicker convergence. An import routing policy is configured on Router R1, which sets a high priority to routes connecting to Router R3. Routes matching 203.0.113.3/32 are installed first because they have a priority of high. LDP imports routes and their configure priority from IS-IS. This route is restored first in the event of a network topology change.



Figure 16: Example: Configuring a Routing Policy to prioritize IS-IS Routes



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 195](#)
- [Configuring Router R1 | 198](#)
- [Results | 200](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter commit from configuration mode.

#### Router R1

```
set interfaces ge-1/0/1 unit 0 description R1->R2
set interfaces ge-1/0/1 unit 0 family inet address 192.0.2.1/24
set interfaces ge-1/0/1 unit 0 family iso
set interfaces ge-1/0/1 unit 0 family inet6 address 2001:db8:1:2::1/64
set interfaces ge-1/0/1 unit 0 family mpls
set interfaces ge-5/0/9 unit 0 description R1->R2
set interfaces ge-5/0/9 unit 0 family inet address 192.0.10.1/24
set interfaces ge-5/0/9 unit 0 family iso
set interfaces ge-5/0/9 unit 0 family inet6 address 2001:db8:1:1::1/64
set interfaces ge-5/0/9 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.1/32
```

```

set interfaces lo0 unit 0 family iso address 49.0002.0103.0000.0010.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:1::1/128
set interfaces lo0 unit 0 family mpls
set protocols mpls ipv6-tunneling
set protocols mpls interface ge-5/0/9.0
set protocols mpls interface ge-1/0/1.0
set protocols isis level 1 disable
set protocols isis interface ge-1/0/1.0
set protocols isis interface ge-5/0/9.0
set protocols isis interface lo0.0 passive
set protocols ldp interface ge-1/0/1.0
set protocols ldp interface ge-5/0/9.0
set protocols ldp interface lo0.0
set policy-options policy-statement test_rf term t1 from route-filter 203.0.113.3/32 exact
set policy-options policy-statement test_rf term t1 then priority high
set protocols isis import test_rf
set routing-options router-id 203.0.113.1
set routing-options autonomous-system 64496

```

## Router R2

```

set interfaces ge-0/0/1 unit 0 description R2->R1
set interfaces ge-0/0/1 unit 0 family inet address 192.0.2.2/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family inet6 address 2001:db8:1:2::2/64
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces ge-0/0/7 unit 0 description R2->R3
set interfaces ge-0/0/7 unit 0 family inet address 198.0.2.1/24
set interfaces ge-0/0/7 unit 0 family iso
set interfaces ge-0/0/7 unit 0 family inet6 address 2001:db8:1:2::1/64
set interfaces ge-0/0/7 unit 0 family mpls
set interfaces ge-1/0/5 unit 0 description R2->R1
set interfaces ge-1/0/5 unit 0 family inet address 192.0.10.2/24
set interfaces ge-1/0/5 unit 0 family iso
set interfaces ge-1/0/5 unit 0 family inet6 address 2001:db8:1:1::2/64
set interfaces ge-1/0/5 unit 0 family mpls
set interfaces ge-1/1/0 unit 0 family inet address 198.0.10.1/24
set interfaces ge-1/1/0 unit 0 family iso
set interfaces ge-1/1/0 unit 0 family inet6 address 2001:db8::1::1/64
set interfaces ge-1/1/0 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0103.0000.0030.00

```

```

set interfaces lo0 unit 0 family inet6 address 2001:db8:2:2::1/128
set interfaces lo0 unit 0 family mpls
set protocols mpls interface ge-1/0/5.0
set protocols mpls interface ge-0/0/1.0
set protocols mpls interface ge-1/1/0.0
set protocols mpls interface ge-0/0/7.0
set protocols isis level 1 disable
set protocols isis interface ge-0/0/1.0
set protocols isis interface ge-0/0/7.0
set protocols isis interface ge-1/0/5.0
set protocols isis interface ge-1/1/0.0
set protocols isis interface lo0.0 passive
set protocols ldp interface ge-0/0/1.0
set protocols ldp interface ge-0/0/7.0
set protocols ldp interface ge-1/0/5.0
set protocols ldp interface ge-1/1/0.0
set protocols ldp interface lo0.0
set routing-options router-id 203.0.113.2
set routing-options autonomous-system 64496

```

### Router R3

```

set interfaces ge-2/0/3 unit 0 description R3->R2
set interfaces ge-2/0/3 unit 0 family inet address 198.1.2.2/24
set interfaces ge-2/0/3 unit 0 family iso
set interfaces ge-2/0/3 unit 0 family inet6 address 2001:db8:1:2::2/64
set interfaces ge-2/0/3 unit 0 family mpls
set interfaces ge-3/0/10 unit 0 description R3->R4
set interfaces ge-3/0/0 unit 0 family inet address 198.0.1.2/24
set interfaces ge-3/0/0 unit 0 family iso
set interfaces ge-3/0/0 unit 0 family inet6 address 2001:db8::1:1::2/64
set interfaces ge-3/0/0 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.3/32
set interfaces lo0 unit 0 family iso address 49.0002.0103.0000.0020.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:3:3::3/128
set interfaces lo0 unit 0 family mpls
set protocols mpls ipv6-tunneling
set protocols mpls interface ge-3/0/0.0
set protocols mpls interface ge-2/0/3.0
set protocols isis level 1 disable
set protocols isis interface ge-2/0/3.0
set protocols isis interface ge-3/0/0.0

```

```

set protocols isis interface lo0.0 passive
set protocols ldp interface ge-2/0/3.0
set protocols ldp interface ge-3/0/0.0
set protocols ldp interface lo0.0
set policy-options policy-statement add_tag term t1 from route-filter 203.0.113.3/32 exact
set policy-options policy-statement add_tag term t1 then tag 18
set protocols isis export add_tag
set routing-options router-id 203.0.113.3
set routing-options autonomous-system 64496

```

## Configuring Router R1

### Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure Router R1:



**NOTE:** Repeat this procedure for other routers after modifying the appropriate interface names, addresses, and other parameters.

1. Configure the interfaces with IPv4 and IPv6 addresses.

```

[edit interfaces]
user@R1# set ge-1/0/1 unit 0 description R1->R2
user@R1# set ge-1/0/1 unit 0 family inet address 192.0.2.1/24
user@R1# set ge-1/0/1 unit 0 family iso
user@R1# set ge-1/0/1 unit 0 family inet6 address 2001:db8:1:2::1/64
user@R1# set ge-1/0/1 unit 0 family mpls
user@R1# set ge-1/0/1 unit 0 description R1->R2
user@R1# set ge-5/0/9 unit 0 family inet address 192.0.10.1/24
user@R1# set ge-5/0/9 unit 0 family iso
user@R1# set ge-5/0/9 unit 0 family inet6 address 2001:db8:1:1::1/64
user@R1# set ge-5/0/9 unit 0 family mpls

```

## 2. Configure the loopback address.

```
[edit interfaces]
user@R1# set lo0 unit 0 family inet address 203.0.113.1/32
user@R1# set lo0 unit 0 family iso address 49.0002.0103.0000.0010.00
user@R1# set lo0 unit 0 family inet6 address 2001:db8:1:1::1/128
user@R1# set lo0 unit 0 family mpls
```

## 3. Configure MPLS.

```
[edit protocols]
user@R1# set mpls ipv6-tunneling
user@R1# set mpls interface ge-5/0/9.0
user@R1# set mpls interface ge-1/0/1.0
```

## 4. Enable IS-IS protocol on the interfaces.

```
[edit protocols]
user@R1# set isis level 1 disable
user@R1# set isis interface ge-1/0/1.0
user@R1# set isis interface ge-5/0/9.0
user@R1# set isis interface lo0.0 passive
user@R1# set isis import test_rf
```

## 5. Configure LDP protocol on the interfaces.

```
[edit protocols]
user@R1# set ldp interface ge-1/0/1.0
user@R1# set ldp interface ge-5/0/9.0
user@R1# set ldp interface lo0.0
```

## 6. Define a policy to prioritize IS-IS routes to Router R3. .

```
[edit policy-options]
user@R1# set policy-statement test_rf term t1 from route-filter 203.0.113.3/32 exact
user@R1# set policy-statement test_rf term t1 then priority high
```

## 7. Configure the router ID and autonomous system (AS) number.

```
[edit routing-options]
user@R1# set routing-options router-id 203.0.113.1
user@R1# set routing-options autonomous-system 64496
```

### Results

From configuration mode, confirm your configuration by entering the **show interfaces**, **show protocols**, **show policy-options** and **show routing-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@R1> show interfaces
ge-1/0/1 {
  unit 0 {
    family inet {
      address 192.0.2.1/24;
    }
    family iso;
    family inet6 {
      address 2001:db8:1:2::1/64;
    }
    family mpls;
  }
}
ge-5/0/9 {
  unit 0 {
    family inet {
      address 192.0.2.1/24;
    }
    family iso;
    family inet6 {
      address 2001:db8:1:1::1/64;
    }
    family mpls;
  }
}
lo0 {
  unit 0 {
    family inet {
```

```

        address 203.0.113.1/32;
    }
    family iso {
        address 49.0002.0103.0000.0010.00;
    }
    family inet6 {
        address 2001:db8:1:1::1/128;
    }
    family mpls;
}
}

```

```

[edit]
user@R1> show protocols
mpls {
    ipv6-tunneling;
    interface ge-5/0/9.0;
    interface ge-1/0/1.0;
}
isis {
    import test_rf;
    level 1 disable;
    interface ge-1/0/1.0;
    interface ge-5/0/9.0;
    interface lo0.0 {
        passive;
    }
}
ldp {
    interface ge-1/0/1.0;
    interface ge-5/0/9.0;
    interface lo0.0;
}

```

```

[edit]
user@R1> show routing-options

```

```
router-id 203.0.113.1;
autonomous-system 64496;
```

```
user@R1> sshow configuration policy-options
policy-statement test_rf {
  term t1 {
    from {
      route-filter 203.0.113.3/32 exact;
    }
    then priority high;
  }
}
```

## Verification

### IN THIS SECTION

- [Verifying the Priority for LDP Routes | 202](#)
- [Verifying the Priority of IS-IS Routes | 204](#)

## Verifying the Priority for LDP Routes

### Purpose

Verify that LDP has inherits route 203.0.113.3 from IS-IS protocol.

### Action

From operational mode, run the **show route extensive** command on Router R1.

```
user@R1> show route 203.0.113.3 extensive
inet.0: 26 destinations, 26 routes (26 active, 0 holddown, 0 hidden)

203.0.113.3/32 (1 entry, 1 announced)
    State: <FlashAll>
TSI:
KRT in-kernel 203.0.113.3/32 -> {16.1.2.2}
```



```

*IS-IS Preference: 18
      Level: 2
      Next hop type: Router, Next hop index: 0
      Address: 0x4a1f43c
      Next-hop reference count: 4
      Next hop: 16.1.2.2 via ge-1/0/1.0, selected
      Session Id: 0x0
      Next hop: 16.1.1.2 via ge-5/0/9.0
      Session Id: 0x0
      State: <Active Int HighPriority>
      Local AS: 64496
      Age: 59      Metric: 20
      Validation State: unverified
      ORR Generation-ID: 0
      Tag: 18
      Task: IS-IS
      Announcement bits (2): 0-KRT 4-LDP
      AS path: I

```

```
inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
```

```
203.0.113.3/32 (1 entry, 1 announced)
```

```

State: <FlashAll>
*LDP Preference: 9
      Next hop type: Router, Next hop index: 0
      Address: 0x4a1e55c
      Next-hop reference count: 2
      Next hop: 16.1.2.2 via ge-1/0/1.0, selected
      Label operation: Push 299776
      Label TTL action: prop-ttl
      Load balance label: Label 299776: None;
      Label element ptr: 0x4b99100
      Label parent element ptr: 0x0
      Label element references: 2
      Label element child references: 0
      Label element lsp id: 0
      Session Id: 0x0
      Next hop: 16.1.1.2 via ge-5/0/9.0
      Label operation: Push 299776
      Label TTL action: prop-ttl
      Load balance label: Label 299776: None;
      Label element ptr: 0x4b99100
      Label parent element ptr: 0x0

```

```

Label element references: 2
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
State:<Active Int HighPriority>
Local AS: 64496
Age: 59      Metric: 1
Validation State: unverified
Task: LDP
Announcement bits (1): 2-Resolve tree 1
AS path: I
Secondary Tables: inet6.3

```

### Meaning

The output shows that LDP inherits the route 203.0.113.3, with priority high from IS-IS.

### Verifying the Priority of IS-IS Routes

#### Purpose

Verify that the priority is set for route 203.0.113.3 in IS-IS.

#### Action

```

user@R1> show isis route download-priority
IS-IS routing table          Current version: L1: 0 L2: 122
IPv4/IPv6 Routes
-----

```

Prefix	L Version	Metric	Type	Interface	NH	Via	Backup	Score
<b>203.0.113.3/32</b>	<b>2</b>	<b>122</b>	<b>20 int</b>	<b>ge-1/0/1.0</b>		<b>IPV4 R2</b>		
				<b>ge-5/0/9.0</b>		<b>IPV4 R2</b>		
203.0.113.2/32	2	122	10 int	ge-1/0/1.0		IPV4 R2		
				ge-5/0/9.0		IPV4 R2		
198.1.1.0/24	2	122	20 int	ge-1/0/1.0		IPV4 R2		
				ge-5/0/9.0		IPV4 R2		
198.1.2.0/24	2	122	20 int	ge-1/0/1.0		IPV4 R2		
				ge-5/0/9.0		IPV4 R2		
2001:db8:2:2::1/128	2	122	10 int	ge-1/0/1.0		IPV6 R2		
				ge-5/0/9.0		IPV6 R2		
2001:db8:3:3::3/128	2	122	20 int	ge-1/0/1.0		IPV6 R2		

				ge-5/0/9.0	IPV6 R2
2001:db8:1:1::/64	2	122	20 int	ge-1/0/1.0	IPV6 R2
				ge-5/0/9.0	IPV6 R2
2001:db8:1:2::/64	2	122	20 int	ge-1/0/1.0	IPV6 R2
				ge-5/0/9.0	IPV6 R2

Meaning

The routes are displayed in the order of the set priorities. Route 203.0.113.3, which is set with high priority is displayed at the very top followed by routes with medium or low priority.

RELATED DOCUMENTATION

<i>import (Protocols IS-IS)</i>
<i>show isis route download priority</i>
<i>show isis route</i>

Configuring Overloading of Stub Networks

Starting in Junos OS Release 18.3R1, new configuration options `external-prefixes` and `internal-prefixes` are available at the `[edit protocols isis overload]` hierarchy level to control overload of internal prefixes, external prefixes or both internal and external prefixes as per network requirement. The user can choose not to receive any traffic for internal and external prefixes advertised by the overloaded IS-IS routers unless the router is the only node in the network which hosts the prefix. In previous Junos OS releases, overloaded IS-IS routers continued to receive traffic for prefixes even if the user did not want to receive traffic for directly connected prefixes.

# Configuring IS-IS Bidirectional Forwarding Detection

## IN THIS CHAPTER

- Understanding BFD for IS-IS | 206
- Example: Configuring BFD for IS-IS | 209
- Understanding BFD Authentication for IS-IS | 217
- Configuring BFD Authentication for IS-IS | 219
- Example: Configuring BFD Authentication for IS-IS | 224

## Understanding BFD for IS-IS

The Bidirectional Forwarding Detection (BFD) protocol is a simple hello mechanism that detects failures in a network. Hello packets are sent at a specified, regular interval. A neighbor failure is detected when the routing device stops receiving a reply after a specified interval. BFD works with a wide variety of network environments and topologies. The failure detection timers for BFD have shorter time limits than the failure detection mechanisms of IS-IS, providing faster detection.

The BFD failure detection timers are adaptive and can be adjusted to be faster or slower. For example, the timers can adapt to a higher value if the adjacency fails, or a neighbor can negotiate a higher value for a timer than the configured value. The timers adapt to a higher value when a BFD session flap occurs more than three times in a span of 15 seconds. A back-off algorithm increases the receive (RX) interval by two if the local BFD instance is the reason for the session flap. The transmission (TX) interval is increased by two if the remote BFD instance is the reason for the session flap.

You can use the `clear bfd adaptation` command to return BFD interval timers to their configured values. The `clear bfd adaptation` command is hitless, meaning that the command does not affect traffic flow on the routing device.



**NOTE:** You can configure IS-IS BFD sessions for IPv6 by including the `bfd-liveness-detection` statement at the `[edit protocols isis interface interface-name family inet|inet6]` hierarchy level.

- For interfaces that support both IPv4 and IPv6 routing, the `bfd-liveness-detection` statement must be configured separately for each inet family.
- BFD over IPv6 link local address is currently not distributed because IS-IS uses link local addresses for forming adjacencies.
- BFD sessions over IPv6 must not have the same aggressive detection intervals as IPv4 sessions.
- BFD IPv6 sessions with detection intervals less than 2.5 seconds are currently not supported when nonstop active routing (NSR) is enabled.

To detect failures in the network, the set of statements in [Table 2 on page 207](#) are used in the configuration.

**Table 2: Configuring BFD for IS-IS**

Statement	Description
<code>bfd-liveness-detection</code>	Enable failure detection.

Table 2: Configuring BFD for IS-IS (*Continued*)

Statement	Description
<code>minimum-interval milliseconds</code>	<p>Specify the minimum transmit and receive intervals for failure detection.</p> <p>This value represents the minimum interval at which the local router transmits hellos packets as well as the minimum interval at which the router expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a number from 1 through 255,000 milliseconds. You can also specify the minimum transmit and receive intervals separately.</p> <p><b>NOTE:</b> Depending on your network environment, these additional recommendations might apply:</p> <ul style="list-style-type: none"> <li>• The recommended minimum interval for centralised BFD is 300 ms with a multiplier of 3, and the recommended minimum interval for distributed BFD is 100 ms with a multiplier of 3.</li> <li>• For very large-scale network deployments with a large number of BFD sessions, please contact Juniper Networks customer support for more information.</li> <li>• For BFD sessions to remain up during a Routing Engine switchover event when <i>nonstop active routing</i> (NSR) is configured, specify a minimum interval of 2500 ms for Routing Engine-based sessions. For distributed BFD sessions with nonstop active routing configured, the minimum interval recommendations are unchanged and depend only on your network deployment.</li> </ul>
<code>minimum-receive-interval milliseconds</code>	<p>Specify only the minimum receive interval for failure detection.</p> <p>This value represents the minimum interval at which the local router expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a number from 1 through 255,000 milliseconds.</p>
<code>multiplier number</code>	<p>Specify the number of hello packets not received by the neighbor that causes the originating interface to be declared down.</p> <p>The default is 3, and you can configure a value from 1 through 225.</p>
<code>no-adaptation</code>	<p>Disable BFD adaptation.</p> <p>You can specify that the BFD sessions not adapt to changing network conditions.</p> <p><b>NOTE:</b> We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.</p>

Table 2: Configuring BFD for IS-IS (*Continued*)

Statement	Description
threshold	<p>Specify the threshold for the following:</p> <ul style="list-style-type: none"> <li>Adaptation of the detection time</li> </ul> <p>When the BFD session detection time adapts to a value equal to or greater than the threshold, a single trap and a system log message are sent.</p> <ul style="list-style-type: none"> <li>Transmit interval</li> </ul> <p><b>NOTE:</b> The threshold value must be greater than the minimum transmit interval multiplied by the multiplier number.</p>
transmit-interval minimum-interval	<p>Specify the minimum transmit interval for failure detection.</p> <p>This value represents the minimum interval at which the local routing device transmits hello packets to the neighbor with which it has established a BFD session. You can configure a value from 1 through 255,000 milliseconds.</p>
version	<p>Specify the BFD version used for detection.</p> <p>The default is to have the version detected automatically.</p>



**NOTE:** You can trace BFD operations by including the traceoptions statement at the [edit protocols bfd] hierarchy level.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

## Example: Configuring BFD for IS-IS

### IN THIS SECTION

- Requirements | 210
- Overview | 210

- Configuration | 211
- Verification | 215

This example describes how to configure the Bidirectional Forwarding Detection (BFD) protocol to detect failures in an IS-IS network.



**NOTE:** BFD is not supported with ISIS for IPV6 on QFX10000 series switches.

## Requirements

Before you begin, configure IS-IS on both routers. See ["Example: Configuring IS-IS" on page 15](#) for information about the required IS-IS configuration.



**NOTE:** We provide the IS-IS configuration in the CLI quick configuration section but do not cover the IS-IS configuration in the step-by-step.

This example uses the following hardware and software components:

- Junos OS Release 7.3 or later
  - Updated and revalidated using Junos OS Release 22.4
- M Series, MX Series, and T Series routers

## Overview

### IN THIS SECTION

- Topology | 210

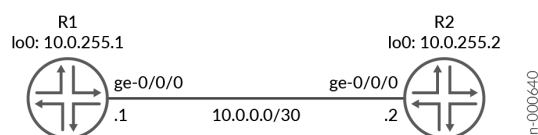
This example shows two routers connected to each other. A loopback interface is configured on each router. IS-IS and BFD protocols are configured on both routers.

## Topology

[Figure 17 on page 211](#) shows the sample network.



Figure 17: Configuring BFD for IS-IS



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 211](#)
- [Procedure | 212](#)
- [Results | 214](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Router R1

```
set interfaces ge-0/0/0 unit 0 family inet address 10.0.0.1/30
set interfaces ge-0/0/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 10.0.255.1/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0255.0001.00
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection version automatic
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection minimum-interval 200
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection minimum-receive-
interval 100
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection multiplier 2
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection no-adaptation
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection transmit-interval
minimum-interval 100
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection transmit-interval
threshold 300
```

```
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection detection-time
threshold 500
set protocols isis interface lo0.0
```

## Router R2

```
set interfaces ge-0/0/0 unit 0 family inet address 10.0.0.2/30
set interfaces ge-0/0/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 10.0.255.2/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0255.0002.00
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection version automatic
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection minimum-interval 200
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection minimum-receive-
interval 100
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection multiplier 2
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection no-adaptation
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection transmit-interval
minimum-interval 100
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection transmit-interval
threshold 300
set protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection detection-time
threshold 500
set protocols isis interface lo0.0
```

## Procedure

### Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode*.



**NOTE:** To simply configure BFD for IS-IS, only the `minimum-interval` statement is required. The BFD protocol selects default parameters for all the other configuration statements when you use the `bfd-liveness-detection` statement without specifying any parameters.



**NOTE:** You can change parameters at any time without stopping or restarting the existing session. BFD automatically adjusts to the new parameter value. However, no

changes to BFD parameters take place until the values resynchronize with each BFD peer.

To configure BFD for IS-IS on Routers R1 and R2:



**NOTE:** We are only showing the steps for R1.

1. Configure the threshold for the adaptation of the detection time, which must be greater than the multiplier number multiplied by the minimum interval.

```
[edit protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection]
user@R1# set detection-time threshold 500
```

2. Configure the minimum transmit and receive intervals for failure detection.

```
[edit protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection]
user@R1# set minimum-interval 200
```

3. Configure only the minimum receive interval for failure detection.

```
[edit protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection]
user@R1# set minimum-receive-interval 100
```

4. Disable BFD adaptation.

```
[edit protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection]
user@R1# set no-adaptation
```

5. Configure the threshold for the transmit interval, which must be greater than the minimum transmit interval.

```
[edit protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection]
user@R1# set transmit-interval threshold 300
```

6. Configure the minimum transmit interval for failure detection.

```
[edit protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection]
user@R1# set transmit-interval minimum-interval 100
```

7. Configure the multiplier number, which is the number of hello packets not received by the neighbor that causes the originating interface to be declared down.

```
[edit protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection]
user@R1# set multiplier 2
```

8. Configure the BFD version used for detection.

The default is to have the version detected automatically.

```
[edit protocols isis interface ge-0/0/0.0 family inet bfd-liveness-detection]
user@R1# set version automatic
```

## Results

From configuration mode, confirm your configuration by issuing the `show protocols isis interface` command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show protocols isis interface ge-0/0/0.0 family inet
bfd-liveness-detection {
  version automatic;
  minimum-interval 200;
  minimum-receive-interval 100;
  multiplier 2;
  no-adaptation;
  transmit-interval {
    minimum-interval 100;
    threshold 300;
  }
  detection-time {
    threshold 500;
```

```
}
}
```

## Verification

### IN THIS SECTION

- [Verifying the Connection Between Routers R1 and R2 | 215](#)
- [Verifying That IS-IS Is Configured | 216](#)
- [Verifying That BFD Is configured | 216](#)

Confirm that the configuration is working properly.

### Verifying the Connection Between Routers R1 and R2

#### Purpose

Make sure that Routers R1 and R2 can reach each other.

#### Action

Ping the other router to check the connectivity between the two routers as per the network topology.

```
user@R1> ping 10.0.0.2 count 2
PING 10.0.0.2 (10.0.0.2): 56 data bytes
64 bytes from 10.0.0.2: icmp_seq=0 ttl=64 time=2.148 ms
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=1.923 ms

--- 10.0.0.2 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.923/2.035/2.148/0.113 ms
```

#### Meaning

Routers R1 and R2 are able to ping each other.

## Verifying That IS-IS Is Configured

### Purpose

Make sure that the IS-IS instance is running on both routers.

### Action

Use the `show isis database` statement to check if the IS-IS instance is running on both routers, R1 and R2.

```
user@R1> show isis database
IS-IS level 1 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R1.00-00              0x1b   0xa2d5      552 L1 L2
R1.02-00              0x2b   0x8da3      545 L1 L2
R2.00-00              0x1a   0x628d      543 L1 L2
  3 LSPs

IS-IS level 2 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R1.00-00              0x1e   0xb9ba      552 L1 L2
R1.02-00              0x2b   0x8da3      545 L1 L2
R2.00-00              0x1d   0x877e      543 L1 L2
  3 LSPs
```

### Meaning

IS-IS is configured on both routers, R1 and R2.

## Verifying That BFD Is configured

### Purpose

Make sure that the BFD instance is running on both routers, R1 and R2.

## Action

Use the `show bfd session detail` statement to check if BFD instance is running on the routers.

```
user@R1> show bfd session detail
```

Address	State	Interface	Detect Time	Transmit Interval	Multiplier
10.0.0.2	Up	ge-0/0/0.0	0.200	0.100	2

Client ISIS L1, TX interval 0.100, RX interval 0.100  
 Client ISIS L2, TX interval 0.100, RX interval 0.100  
 Session up time 00:02:41, previous down time 00:00:09  
 Local diagnostic None, remote diagnostic None  
 Remote state Up, version 1  
 Session type: Single hop BFD

1 sessions, 2 clients  
 Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps

## Meaning

BFD is configured on Routers R1 and R2 for detecting failures in the IS-IS network.

## Understanding BFD Authentication for IS-IS

### IN THIS SECTION

- [BFD Authentication Algorithms | 218](#)
- [Security Authentication Keychains | 219](#)
- [Strict Versus Loose Authentication | 219](#)

Bidirectional Forwarding Detection (BFD) enables rapid detection of communication failures between adjacent systems. By default, authentication for BFD sessions is disabled. However, when running BFD over Network Layer protocols, the risk of service attacks can be significant. We strongly recommend using authentication if you are running BFD over multiple hops or through insecure tunnels. Junos OS

supports authentication for BFD sessions running over IS-IS. BFD authentication is only supported in the domestic image and is not available in the export image.

You authenticate BFD sessions by specifying an authentication algorithm and keychain, and then associating that configuration information with a security authentication keychain using the keychain name.

The following sections describe the supported authentication algorithms, security keychains, and level of authentication that can be configured:

## BFD Authentication Algorithms

Junos OS supports the following algorithms for BFD authentication:

- **simple-password**—Plain-text password. One to 16 bytes of plain text are used to authenticate the BFD session. One or more passwords might be configured. This method is the least secure and should be used only when BFD sessions are not subject to packet interception.
- **keyed-md5**—Keyed Message Digest 5 hash algorithm for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed MD5 uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than or equal to the last sequence number received. Although more secure than a simple password, this method is vulnerable to replay attacks. Increasing the rate at which the sequence number is updated can reduce this risk.
- **meticulous-keyed-md5**—Meticulous keyed Message Digest 5 hash algorithm. This method works in the same manner as keyed MD5, but the sequence number is updated with every packet. Although more secure than keyed MD5 and simple passwords, this method might take additional time to authenticate the session.
- **keyed-sha-1**—Keyed Secure Hash Algorithm I for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed SHA uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. The key is not carried within the packets. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than the last sequence number received.
- **meticulous-keyed-sha-1**—Meticulous keyed Secure Hash Algorithm I. This method works in the same manner as keyed SHA, but the sequence number is updated with every packet. Although more secure than keyed SHA and simple passwords, this method might take additional time to authenticate the session.





**NOTE:** *Nonstop active routing* (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algorithms might go down after a switchover.

## Security Authentication Keychains

The security authentication keychain defines the authentication attributes used for authentication key updates. When the security authentication keychain is configured and associated with a protocol through the keychain name, authentication key updates can occur without interrupting routing and signaling protocols.

The authentication keychain contains one or more keychains. Each keychain contains one or more keys. Each key holds the secret data and the time at which the key becomes valid. The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

BFD allows multiple clients per session, and each client can have its own keychain and algorithm defined. To avoid confusion, we recommend specifying only one security authentication keychain.

## Strict Versus Loose Authentication

By default, strict authentication is enabled and authentication is checked at both ends of each BFD session. Optionally, to smooth migration from nonauthenticated sessions to authenticated sessions, you can configure *loose checking*. When loose checking is configured, packets are accepted without authentication being checked at each end of the session. This feature is intended for transitional periods only.

### RELATED DOCUMENTATION

| [Example: Configuring BFD Authentication for IS-IS](#) | 224

## Configuring BFD Authentication for IS-IS

### IN THIS SECTION

● [Configuring BFD Authentication Parameters](#) | 220

Configure authentication for BFD sessions running over IS-IS. Routing instances are also supported. Only three steps are needed to configure authentication on a BFD session:

1. Specify the BFD authentication algorithm for the IS-IS protocol.
2. Associate the authentication keychain with the IS-IS protocol.
3. Configure the related security authentication keychain.

The following sections provide instructions for configuring and viewing BFD authentication on IS-IS:

## Configuring BFD Authentication Parameters

To configure BFD authentication:

1. Specify the algorithm (**keyed-md5**, **keyed-sha-1**, **meticulous-keyed-md5**, **meticulous-keyed-sha-1**, or **simple-password**) to use for BFD authentication on an IS-IS route or routing instance.

```
[edit]
user@host# set protocols isis interface ge-4/0/0.0 bfd-liveness-detection authentication
algorithm keyed-sha-1
```



**NOTE:** Nonstop active routing (NSR) is not supported with the meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algorithms might go down after a switchover.

2. Specify the keychain to be used to associate BFD sessions on the specified IS-IS route or routing instance with the unique security authentication keychain attributes.

This should match the keychain name configured at the [edit security authentication key-chains] hierarchy level.

```
[edit]
user@host# set protocols isis interface ge-4/0/0.0 bfd-liveness-detection authentication
keychain bfd-isis
```



**NOTE:** The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

3. Specify the unique security authentication information for BFD sessions:

- The matching keychain name as specified in Step 2.
- At least one key, a unique integer between **0** and **63**. Creating multiple keys allows multiple clients to use the BFD session.
- The secret data used to allow access to the session.
- The time at which the authentication key becomes active, *yyyy-mm-dd.hh:mm:ss*.

```
[edit security]
user@host# set authentication-key-chains key-chain bfd-sr4 key 53 secret $9$ggaJDmPQ6/tJgF/
AtREVsyPsnCtUhm start-time 2009-06-14.10:00:00
```

4. (Optional) Specify loose authentication checking if you are transitioning from nonauthenticated sessions to authenticated sessions.

```
[edit]
user@host> set protocols isis interface ge-4/0/0.0 bfd-liveness-detection authentication
loose-check
```

5. (Optional) View your configuration using the `show bfd session detail` or `show bfd session extensive` command.

6. Repeat these steps to configure the other end of the BFD session.



**NOTE:** BFD authentication is only supported in the domestic image and is not available in the export image.

## Viewing Authentication Information for BFD Sessions

You can view the existing BFD authentication configuration using the `show bfd session detail` and `show bfd session extensive` commands.

The following example shows BFD authentication configured for the `ge-4/0/0.0` interface. It specifies the keyed SHA-1 authentication algorithm and a keychain name of `bfd-isis`. The authentication keychain is configured with two keys. Key **1** contains the secret data “`$9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUhm`”

and a start time of June 1, 2009, at 9:46:02 AM PST. Key 2 contains the secret data “\$9\$a5jiKW9l.reP38ny.TszF2/9” and a start time of June 1, 2009, at 3:29:20 PM PST.

```
[edit protocols isis]
interface ge-4/0/0.0 {
    bfd-liveness-detection {
        authentication {
            algorithm keyed-sha-1;
            key-chain bfd-isis;
        }
    }
}
[edit security]
authentication key-chains {
    key-chain bfd-isis {
        key 1 {
            secret "$9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUhm";
            start-time "2009-6-1.09:46:02 -0700";
        }
        key 2 {
            secret "$9$a5jiKW9l.reP38ny.TszF2/9";
            start-time "2009-6-1.15:29:20 -0700";
        }
    }
}
```

If you commit these updates to your configuration, you see output similar to the following. In the output for the `show bfd sessions detail` command, **Authenticate** is displayed to indicate that BFD authentication is configured. For more information about the configuration, use the `show bfd sessions extensive` command. The output for this command provides the keychain name, the authentication algorithm and mode for each client in the session, and the overall BFD authentication configuration status, keychain name, and authentication algorithm and mode.

#### show bfd sessions detail

```
user@host# show bfd session detail
```

Address	State	Interface	Detect Time	Transmit Interval	Multiplier
10.9.1.29	Up	ge-4/0/0.0	0.600	0.200	3
Client ISIS L2, TX interval 0.200, RX interval 0.200, multiplier 3, <b>Authenticate</b>					
Session up time 3d 00:34, previous down time 00:00:01					

```
Local diagnostic NbrSignal, remote diagnostic AdminDown
Remote state Up, version 1

1 sessions, 1 clients
Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

**show bfd sessions extensive**

```
user@host# show bfd session extensive
```

Address	State	Interface	Detect Time	Transmit Interval	Multiplier
10.9.1.29	Up	ge-4/0/0.0	0.600	0.200	3

```
Client ISIS L2, TX interval 0.200, RX interval 0.200, multiplier 3, Authenticate
keychain bfd-isis, algo keyed-sha-1, mode strict
Session up time 00:04:42
Local diagnostic None, remote diagnostic NbrSignal
Remote state Up, version 1
Replicated
Min async interval 0.300, min slow interval 1.000
Adaptive async TX interval 0.300, RX interval 0.300
Local min TX interval 0.300, minimum RX interval 0.300, multiplier 3
Remote min TX interval 0.300, min RX interval 0.300, multiplier 3
Local discriminator 2, remote discriminator 2
Echo mode disabled/inactive
Authentication enabled/active, keychain bfd-isis, algo keyed-sha-1, mode strict

1 sessions, 1 clients

Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

**RELATED DOCUMENTATION**

[Understanding BFD Authentication for IS-IS | 217](#)

*Example: Configuring BFD for IS-IS*

*Understanding BFD for IS-IS*

## Example: Configuring BFD Authentication for IS-IS

### IN THIS SECTION

- Requirements | 224
- Overview | 224
- Configuration | 225
- Verification | 228

This example shows how to configure BFD authentication for IS-IS.

### Requirements

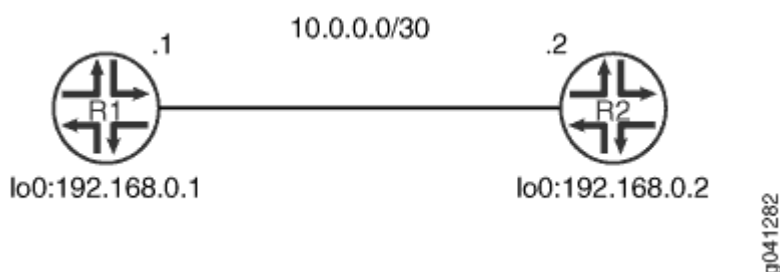
Before you begin, configure IS-IS on both routers. See ["Example: Configuring IS-IS" on page 15](#) for information about the required IS-IS configuration.

### Overview

In this example, a BFD authentication keychain is configured with meticulous keyed MD5 authentication.

[Figure 18 on page 224](#) shows the topology used in this example.

**Figure 18: IS-IS BFD Authentication Topology**



["CLI Quick Configuration" on page 225](#) shows the configuration for both of the devices in [Figure 18 on page 224](#). The section ["No Link Title" on page 226](#) describes the steps on Device R1.

## Configuration

### IN THIS SECTION

- [Procedure | 225](#)

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R1

```
set security authentication-key-chains key-chain secret123 description for-isis-bfd
set security authentication-key-chains key-chain secret123 key 1 secret $ABC123
set security authentication-key-chains key-chain secret123 key 1 start-time "2012-5-31.13:00:00
-0700"
set security authentication-key-chains key-chain secret123 key 2 secret $ABC123
set security authentication-key-chains key-chain secret123 key 2 start-time "2013-5-31.13:00:00
-0700"
set security authentication-key-chains key-chain secret123 key 3 secret $ABC123
set security authentication-key-chains key-chain secret123 key 3 start-time "2014-5-31.13:00:00
-0700"
set protocols isis interface ge-1/2/0.0 bfd-liveness-detection minimum-interval 100
set protocols isis interface ge-1/2/0.0 bfd-liveness-detection authentication key-chain secret123
set protocols isis interface ge-1/2/0.0 bfd-liveness-detection authentication algorithm
meticulous-keyed-md5
```

#### Device R2

```
set security authentication-key-chains key-chain secret123 description for-isis-bfd
set security authentication-key-chains key-chain secret123 key 1 secret $ABC123
set security authentication-key-chains key-chain secret123 key 1 start-time "2012-5-31.13:00:00
-0700"
set security authentication-key-chains key-chain secret123 key 2 secret $ABC123
```

```

set security authentication-key-chains key-chain secret123 key 2 start-time "2013-5-31.13:00:00
-0700"
set security authentication-key-chains key-chain secret123 key 3 secret $ABC123
set security authentication-key-chains key-chain secret123 key 3 start-time "2014-5-31.13:00:00
-0700"
set protocols isis interface ge-1/2/0.0 bfd-liveness-detection minimum-interval 100
set protocols isis interface ge-1/2/0.0 bfd-liveness-detection authentication key-chain secret123
set protocols isis interface ge-1/2/0.0 bfd-liveness-detection authentication algorithm
meticulous-keyed-md5

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure IS-IS BFD authentication:

1. Configure the authentication keychain.

```

[edit security authentication-key-chains key-chain secret123]
user@R1# set description for-isis-bfd
user@R1# set key 1 secret "$ABC123"
user@R1# set key 1 start-time "2012-5-31.13:00:00 -0700"
user@R1# set key 2 secret "$ABC123"
user@R1# set key 2 start-time "2013-5-31.13:00:00 -0700"
user@R1# set key 3 secret "$ABC123"
user@R1# set key 3 start-time "2014-5-31.13:00:00 -0700"

```

2. Enable BFD.

```

[edit protocols isis interface ge-1/2/0.0 bfd-liveness-detection]
user@R1# set minimum-interval 100

```

3. Apply the authentication keychain.

```

[edit protocols isis interface ge-1/2/0.0 bfd-liveness-detection]
user@R1# set authentication key-chain secret123

```



#### 4. Set the authentication type.

```
[edit protocols isis interface ge-1/2/0.0 bfd-liveness-detection]
user@R1# set authentication algorithm meticulous-keyed-md5
```

## Results

From configuration mode, confirm your configuration by entering the `show protocols` and `show security` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show protocols
isis {
  interface ge-1/2/0.0 {
    bfd-liveness-detection {
      minimum-interval 100;
      authentication {
        key-chain secret123;
        algorithm meticulous-keyed-md5;
      }
    }
  }
}
```

```
user@R1# show security
authentication-key-chains {
  key-chain secret123 {
    description for-isis-bfd;
    key 1 {
      secret "$ABC123";
      start-time "2012-5-31.13:00:00 -0700";
    }
    key 2 {
      secret "$ABC123";
      start-time "2013-5-31.13:00:00 -0700";
    }
    key 3 {
      secret "$ABC123";
      start-time "2014-5-31.13:00:00 -0700";
    }
  }
}
```

```
    }  
  }  
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

IN THIS SECTION

Verifying IS-IS BFD Authentication | 228

Confirm that the configuration is working properly.

Verifying IS-IS BFD Authentication

Purpose

Verify the status of IS-IS BFD authentication.

Action

From operational mode, enter the `show bfd session extensive` command.

```
user@R1> show bfd session extensive  
  
Address           State    Interface    Detect   Transmit  
10.0.0.2           Down     ge-1/2/0.0   Time   Interval Multiplier  
0.300             1.000      3  
Client ISIS L1, TX interval 0.100, RX interval 0.100, Authenticate  
keychain secret123, algo meticulous-keyed-md5, mode strict  
Client ISIS L2, TX interval 0.100, RX interval 0.100, Authenticate  
keychain secret123, algo meticulous-keyed-md5, mode strict  
Session down time 00:35:13, previous up time 00:12:17  
Local diagnostic None, remote diagnostic None  
Remote state Up, version 1  
Logical system 2, routing table index 85  
Min async interval 0.100, min slow interval 1.000  
Adaptive async TX interval 0.100, RX interval 0.100  
Local min TX interval 1.000, minimum RX interval 0.100, multiplier 3
```

```
Remote min TX interval 0.100, min RX interval 0.100, multiplier 3
Local discriminator 2, remote discriminator 1
Echo mode disabled/inactive, no-absorb, no-refresh
Authentication enabled/active, keychain secret123, algo meticulous-keyed-md5, mode strict
Session ID: 0x100101
```

```
1 sessions, 2 clients
```

```
Cumulative transmit rate 1.0 pps, cumulative receive rate 10.0 pps
```

## Meaning

The output shows that BFD authentication is enabled on IS-IS Level 1 and Level 2.

## RELATED DOCUMENTATION

| [Configuring BFD Authentication for IS-IS](#) | 219

# Configuring IS-IS Flood Groups

## IN THIS CHAPTER

- [Understanding IS-IS Flood Group | 230](#)
- [Example: Configuring IS-IS Flood Group | 231](#)
- [How to Configure Flood-Reflector Interfaces in IS-IS Networks | 237](#)

## Understanding IS-IS Flood Group

IS-IS supports flood-group. This feature limits link-state packet data unit (PDU) flooding over IS-IS interfaces.

A link-state packet (LSP) that is not self-originated will be flooded only through the interface belonging to the flood group that has the configured area ID in the LSP. This helps minimize the routes and topology information, thus ensuring optimal convergence. You can segregate both Level 1 and Level 2 IS-IS routers into flood groups by using area IDs as tags to identify a flood group. Configure interfaces with specific area IDs to modify the flooding behavior as per your requirements. To enable IS-IS flood group, include the `flood-group flood-group-area-ID` statement at the [edit protocols isis interface] hierarchy level.

## RELATED DOCUMENTATION

[IS-IS Overview | 2](#)

[Example: Configuring IS-IS Flood Group | 231](#)

## Example: Configuring IS-IS Flood Group

### IN THIS SECTION

- [Requirements | 231](#)
- [Overview | 231](#)
- [Configuration | 232](#)
- [Verification | 235](#)

### Requirements

This example uses the following hardware and software components:

- Four routers.
- Junos OS.

Before you begin:

1. Configure IS-IS routing protocol on the routers.
2. Configure IS-IS interfaces with specific area IDs to modify the flood behavior as per your requirements.

### Overview

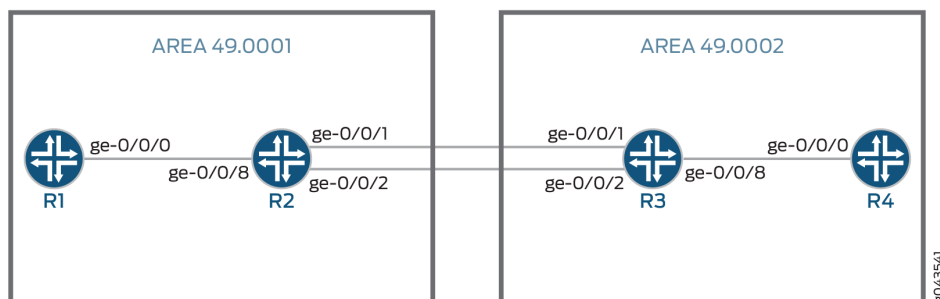
#### IN THIS SECTION

- [Topology | 231](#)

### Topology

In this topology, flood-group is configured on router R2.

Figure 19: Flood Group Topology



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 232](#)
- [Procedure | 234](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### R1

```
set interfaces ge-0/0/0 description "Connected To R2"
set interfaces ge-0/0/0 unit 0 family inet address 81.1.3.3/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 81.3.3.3/32
set interfaces lo0 unit 0 family iso address 49.0001.0810.0300.3003.00
set protocols isis interface all
set protocols isis interface fxp0.0 disable
```

#### R2

```
set interfaces ge-0/0/1 description "Connected To R3"
set interfaces ge-0/0/1 unit 0 family inet address 30.1.1.1/32
```

```

set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/2 description "Connected To R3"
set interfaces ge-0/0/2 unit 0 family inet address 40.1.1.1/32
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/8 description "Connected To R1"
set interfaces ge-0/0/8 unit 0 family inet address 81.1.3.1/24
set interfaces ge-0/0/8 unit 0 family iso
set interfaces lo0 unit 0 family inet address 81.1.1.1/32
set interfaces lo0 unit 0 family iso address 49.0001.0810.0100.1001.00
set protocols isis interface all
set protocols isis interface fxp0.0 disable

```

### R3

```

set interfaces ge-0/0/1 description "Connected To R2"
set interfaces ge-0/0/1 unit 0 family inet address 30.1.1.2/32
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/2 description "Connected To R2"
set interfaces ge-0/0/2 unit 0 family inet address 40.1.1.2/32
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/8 description "Connected To R4"
set interfaces ge-0/0/8 unit 0 family inet address 81.2.4.2/24
set interfaces ge-0/0/8 unit 0 family iso
set interfaces lo0 unit 0 family inet address 81.2.2.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0810.0200.2002.00
set protocols isis interface all
set protocols isis interface fxp0.0 disable

```

### R4

```

set interfaces ge-0/0/0 description "Connected To R3"
set interfaces ge-0/0/0 unit 0 family inet address 81.2.4.4/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 81.4.4.4/32
set interfaces lo0 unit 0 family iso address 49.0002.0810.0400.4004.00
set protocols isis interface all
set protocols isis interface fxp0.0 disable

```



**NOTE:** 1. commit after every configuration.

2. Following is the output before configuring *flood-group* on R2. You will see the link-state packets (LSPs) of R1, R2, R3 and R4.

From operational mode, run the `show isis database` command on router R1.

```
user@R1> show isis database
```

```
user@R1# run show isis database
```

```
IS-IS level 1 link-state database:
```

LSP ID	Sequence	Checksum	Lifetime	Attributes
R2.00-00	0x3	0xea11	1175	L1 L2
R1.00-00	0x5	0x34f8	1197	L1 L2
R1.02-00	0x2	0	0	L1 L2

3 LSPs

```
IS-IS level 2 link-state database:
```

LSP ID	Sequence	Checksum	Lifetime	Attributes
R2.00-00	0x1b	0x2ccc	1175	L1 L2
R3.00-00	0x21	0xb15e	865	L1 L2
R3.02-00	0xb	0xdac3	839	L1 L2
R3.03-00	0xc	0xd1ca	865	L1 L2
R3.04-00	0x8	0x33ff	618	L1 L2
R1.00-00	0xb	0x2cfa	1197	L1 L2
R1.02-00	0x8	0	0	L1 L2
R4.00-00	0xc	0x40c3	621	L1 L2

## Procedure

### Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [Junos OS CLI User Guide](#).

Flood-group functionality check:



### 1. Deactivate protocol IS-IS on routers R1 and R2

```
[edit protocols]
user@R1# deactivate protocols isis
user@R1# commit
```

```
[edit protocols]
user@R2# deactivate protocols isis
user@R2# commit
```

### 2. Configure flood-group on interface of router R2: set protocol isis interface *interface* flood-group *flood-group-area-ID*

```
[edit protocols]
user@R2# set protocols isis interface ge-0/0/8.0 flood-group 49.0001
user@R2# commit
```

### 3. Activate protocol IS-IS on routers R1 and R2 and wait until the adjacency comes up.

```
[edit protocols]
user@R1# activate protocols isis
user@R1# commit
```

```
[edit protocols]
user@R2# activate protocols isis
user@R2# commit
```

## Verification

### IN THIS SECTION


- [Verifying the IS-IS Database | 236](#)

Verifying the IS-IS Database

Purpose

Verify IS-IS database.

Action



**NOTE:** Following is the output after configuring *flood-group* on R2. `show isis database` on router R1 will show LSPs from router R1 and router R2 only. *flood-group* is applicable to non self-originated LSPs only.

From operational mode, run the `show isis database` command on router R1.

```
user@R1> show isis database

user@R1# run show isis database
IS-IS level 1 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R2.00-00              0x2    0x43b9    1123 L1 L2
R1.00-00              0x2    0x8e60    1125 L1 L2
R1.02-00              0x1    0x88e9    1125 L1 L2
  3 LSPs

IS-IS level 2 link-state database:
LSP ID                Sequence Checksum Lifetime Attributes
R2.00-00              0x1a   0x7485    1148 L1 L2
R1.00-00              0x9    0xddaf    1150 L1 L2
R1.02-00              0x1    0x88e9    1150 L1 L2
  3 LSPs
```

RELATED DOCUMENTATION

<a href="#">Understanding IS-IS Flood Group   230</a>
<code>show isis database</code>

## How to Configure Flood-Reflector Interfaces in IS-IS Networks

### SUMMARY

Learn to configure flood-reflector interfaces in IS-IS networks for flooding path reduction, fast convergence, network efficiency, scalability.

### IN THIS SECTION

- [Understanding IS-IS Flood Reflectors | 237](#)
- [Example: IS-IS Flood Reflector | 245](#)

## Understanding IS-IS Flood Reflectors

### IN THIS SECTION

- [Benefits of Flood Reflectors | 237](#)
- [Flood Reflectors Overview | 237](#)
- [Junos OS Implementation of Flood Reflectors | 238](#)
- [Limitations | 242](#)

### Benefits of Flood Reflectors

- **Flooding path reduction**—Reduces the redundancy in flooding paths as it limits flooding of link-state packet data units (PDUs) and enhances the efficiency of IS-IS updates in large fabric topologies.
- **Fast convergence**—Optimizes IS-IS routing protocol convergence in large networks.
- **Network efficiency**—Enhances network efficiency rapidly by leveraging the pre-existing capability to establish IS-IS adjacencies over flexible tunnel interfaces (FTI).
- **Scalability**—Provides better scalability for the Level 2 topologies in an IS-IS network. Only the routers configured as flood reflectors participate in flood reflection. This leverages incremental deployment of scalable Level 1 transit areas in an existing network, without the necessity of upgrading other routers in the network.

### Flood Reflectors Overview

A flood-reflector adjacency as defined in the [draft-przygienda-flood-reflector-00](#) is built for the purpose of reflecting flooding information. The flood reflectors participate in the IS-IS control plane without

being used in the forwarding plane. This is a purely local operation on the Level 1/Level 2 ingress device as it does not require replacing or modifying any routers not involved in the reflection process.

IS-IS flood reflection enables creation of flood-reflection topologies where Level 1 areas provide transit forwarding for Level 2 destinations within a Level 2 topology. This is accomplished by creating Level 2 flood-reflection adjacencies within each Level 1 area. The Level 2 flood-reflection adjacencies are used to flood Level 2 link-state PDUs that are used in the Level 2 shortest-path-first (SPF) computation. However, they are not used for forwarding. This arrangement provides better scalability for the Level 2 topology.

To establish IS-IS adjacency for flood reflection, we designate flexible tunnel interfaces (FTI) as flood-reflector interfaces. These tunnels utilize UDP encapsulation.

## Junos OS Implementation of Flood Reflectors

### Overview

In the Junos OS implementation, the basic flood reflector forwarding functionality enables us to identify an IS-IS interface as a flood-reflector interface on a per-level basis. This modifies the Level 2 route computation to not install any next hops that uses a flood-reflector interface as a next hop.

If this process results in at least one remaining next hop that uses a normal interface, then the modified Level 2 route is installed. If the process of removing flood-reflector next hops from the Level 2 route results in a Level 2 route that has no next hops, then the installation of the Level 2 route is suppressed completely. Because the installation of the usual IS-IS Level 2 route is suppressed, we rely on the presence of an IS-IS Level 1 route to carry the traffic to the flood-reflector client on the Level 2 shortest path for this prefix.

Flood reflection does not load balance traffic on Level 2 and Level 1 routes. Suppose a Level 2 route has 10 equal-cost next hops and one of those next hops uses a flood-reflector interface, then all the next hops are removed from the Level 2 route. Even though there is a path available in the Level 2 domain, it suppresses all the Level 2 routes and relies on the IS-IS Level 1/ Level 2 inter-area route to carry the traffic. The Junos OS implementation does not use a Level 2 route until the Level 2 route has at least one flood-reflector next hop.



**NOTE:** Ensure that you have configured the Level 1 routes to prevent disruption of traffic.

### Flood-Reflection Adjacency Formation

The Flood-Reflection TLV as defined in [draft-przygienda-flood-reflector-00](#) is a new top-level TLV that represents the flood-reflector cluster that a given router interface is configured to participate in. It also indicates whether the router is configured to play the role of either the flood reflector or the flood-

reflector client. For more information about the flood reflection TLV, see [draft-przygienda-flood-reflector-00](#)

Flood reflection implements the advertisement and receipt of the flood-reflection adjacency sub-TLV as defined in [draft-przygienda-lsr-flood-reflection-01](#). The flood reflection adjacency sub-TLV is installed in the traffic engineering database (TED) and is included in the Level 2 area flooded LSPs. It indicates that a given adjacency is a flood-reflector adjacency and serves the following purposes:

- It enables RSVP on the same router to recognize that a link in the TED represents a flood-reflection adjacency.
- It also helps in potential metric-independent loop prevention mechanism. That is, it enables a device participating in flood reflection to have awareness of remote flood-reflection links to detect loops.

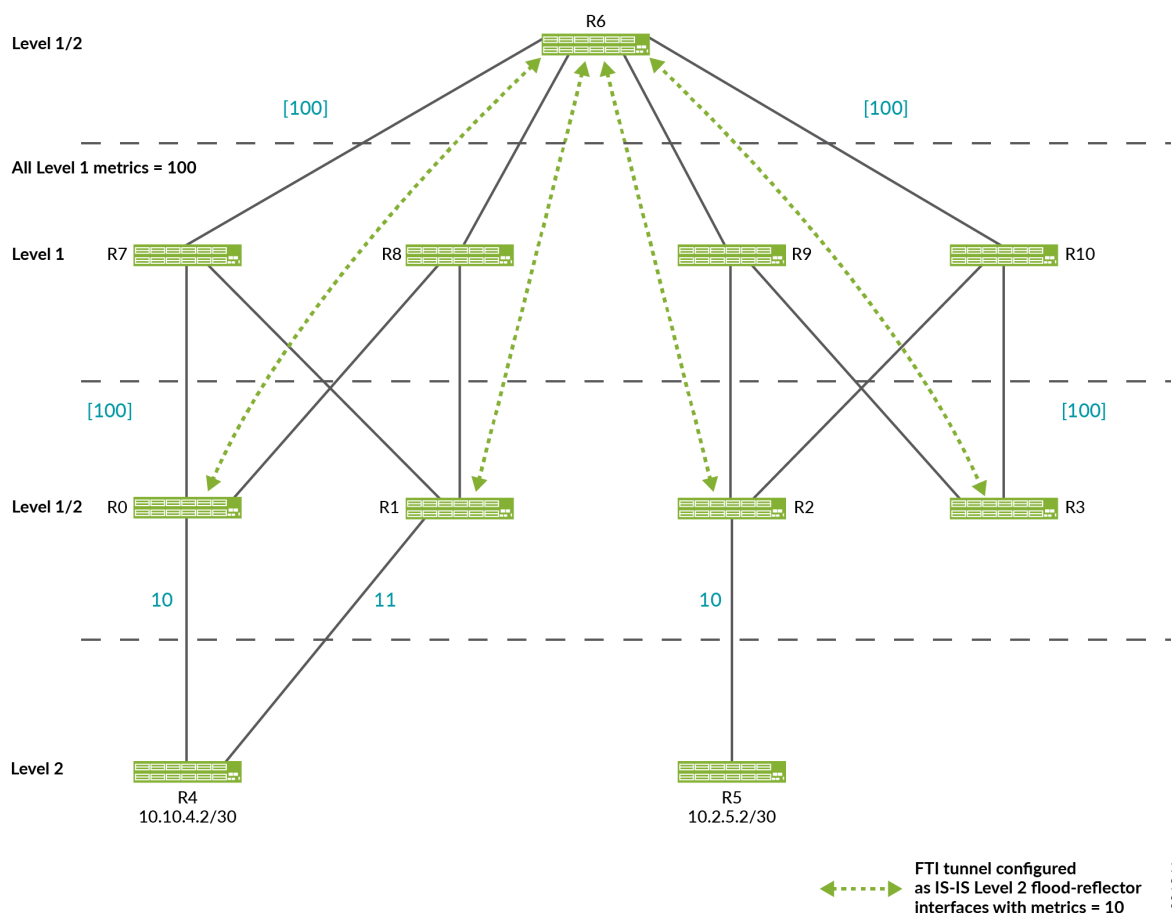


**NOTE:** The external Level 2 devices that do not participate in flood reflection do not advertise or receive the flood-reflection adjacency sub-TLV.

## Flood Reflector Sample Topology

[Figure 20 on page 240](#) depicts a flood-reflector topology that allows R4 to send traffic to R5 utilizing the Level 1 fabric shown at the top of the figure without being exposed to any Level 1 advertisements.

Figure 20: Flood Reflector Sample Topology



- R6 is the flood reflector. R0, R1, R2, R3 are flood-reflector clients and have FTI tunnels to R6. All of the FTI tunnels have metric 10 and are configured as flood-reflector interfaces. R0, R1, R2, R3, R6 are configured to redistribute Level 2 routes into Level 1 as Level 1/Level 2 inter-area routes. This redistribution into Level 1 only occurs if the Level 2 route is installed in the route table.
- R2 advertises 10.2.5.2/30 in Level 1 link-state PDUs with a cost of 10 as it has installed the Level 2 route for 10.2.5.2/30 over the physical interface from R2 to R5.
- Under normal circumstances, if the FTI tunnel from R3 to R6 was not configured as a flood reflector interface, then R3 would also advertise 10.2.5.2/30 into Level 1 link-state PDUs with a cost of 30.

However, as the FTI tunnel from R3 to R6 is configured as a flood reflector-interface, the Level 2 route is suppressed in favor of the Level 1 route for 10.2.5.2/30 being advertised by R3 into Level 1 link-state PDUs. The same logic applies to R0 and R1. Therefore, only R2 advertises 10.2.5.2/24 in Level 1 link-state PDUs.

- When we trace a packet destined for 10.2.5.2/30 from R4 to R5, R4 only sees Level 2 advertisements. Therefore, it determines that the Level 2 shortest path to reach 10.2.5.2/30 is R4-R0-R6-R2-R5. R4 sends the packet to R0.
- At R0, the shortest Level 2 path to reach 10.2.5.2/30 is R0-R6-R2-R5 (with cost 30). However, as the next hop for this Level 2 route uses a flood-reflector interface, the Level 2 route is suppressed. Instead, the Level 1 route to reach R2 is used.
- R7, R8, R9, R10 use the Level 1 route to reach R2 as they do not participate in Level 2. R6 uses the Level 1 route because all of the Level 2 routes on R6 use FTI tunnels configured as Level 2 flood-reflector interfaces, so all Level 2 routes are suppressed at R6.
- At R2, the Level 2 internal route for 10.2.5.2/30 using the physical Level 2 interface to R5 is installed and used.

### Requirements for Configuring Flood-reflector Interfaces

The following requirements are enforced through IS-IS advertisements by including information about the role of the router (flood reflector or flood-reflector client) as well as the cluster-identifier (ID) in the IS-IS Hello messages:

- A flood-reflector client must not be allowed to connect to another flood-reflector client over a flood-reflector interface.
- A flood-reflector client must be allowed to connect to multiple flood reflectors over flood-reflector interfaces.
- A flood reflector must not be allowed to connect to another flood reflector over a flood reflector-interface.
- Adjacency between a flood reflector and a flood reflector client can be established only if they have the same cluster ID.
- A flood reflector for a given level (Level 1, Level 2, or Level 1/Level 2) must not have any IS-IS interfaces at a given level that are not flood-reflector interfaces. This can be validated with `commit check` without any advertisements.



**BEST PRACTICE:** We recommend configuring the Level 2 tunnels to use the source and destination loopback addresses that are only advertised into Level 1. Different loopback addresses are advertised into Level 2. Otherwise, you might encounter a scenario where the edge router in the fabric becomes disconnected from the Level 1 fabric, but it is still able to form a flood-reflector adjacency by tunneling over links in the Level 2 topology.

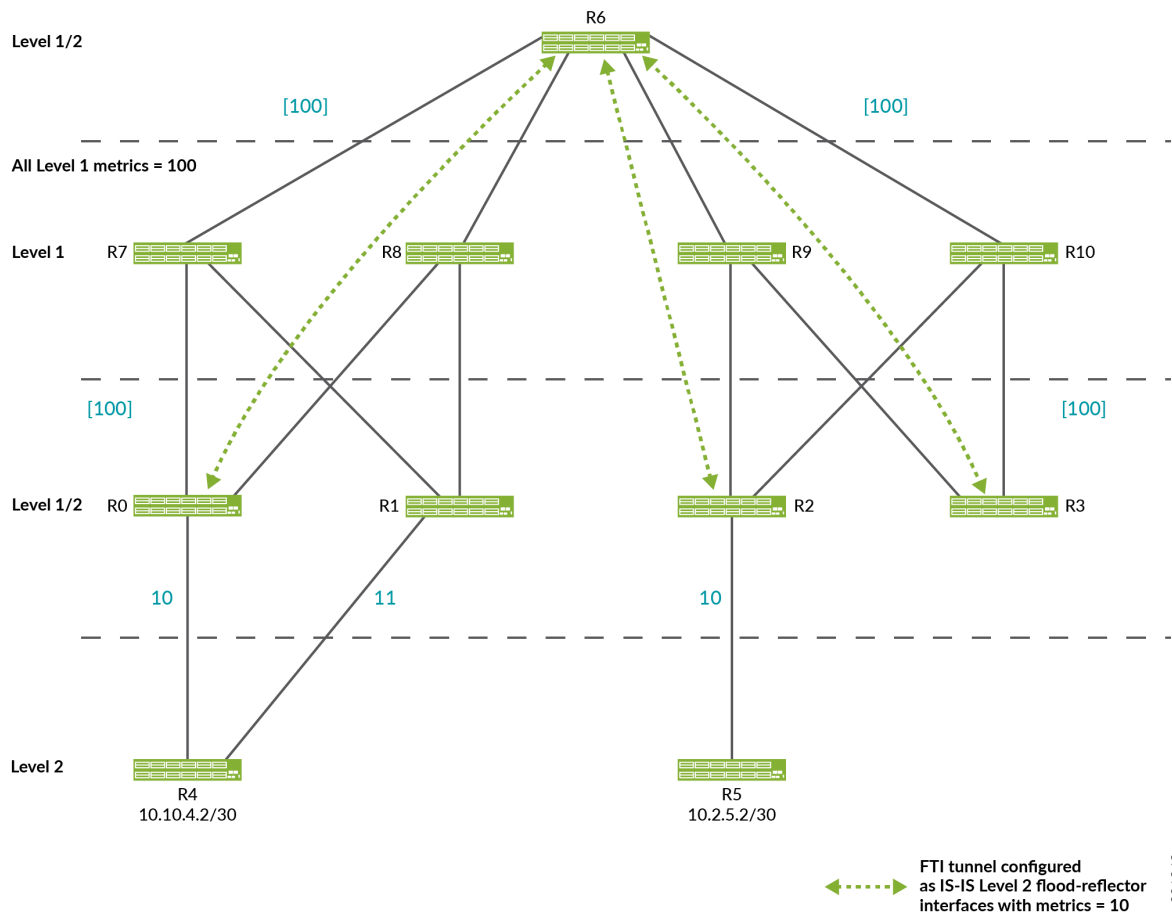
## Limitations

Flood reflection carries with it the potential to disrupt traffic and routing loops in certain topologies where it is not configured properly.

## Routing Loops

When using flood reflection, it is possible to create topologies that cause routing loops. [Figure 21 on page 242](#) depicts a looping topology sample.

**Figure 21: Looping Topology**



The only difference between the topologies in [Figure 20 on page 240](#) and [Figure 21 on page 242](#) is the missing the FTI tunnel between R1 and R6 in [Figure 21 on page 242](#).



## Other Factors that Cause Routing Loops

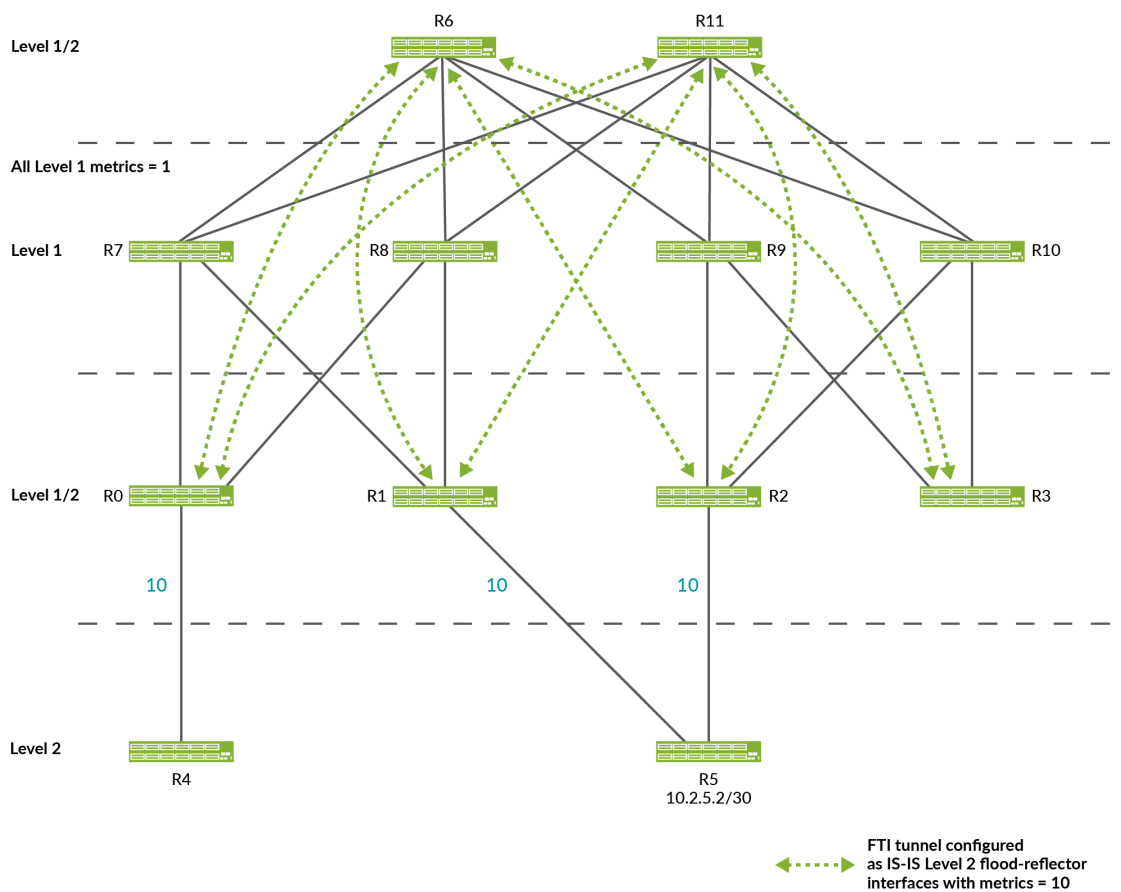
Besides the missing FTI tunnel described in the topology in [Figure 21 on page 242](#), the following factors can also cause a routing loop:

- Tunnel establishment being slow or a tunnel never getting established.
- Level 1 metrics being larger than the Level 2 metrics.

## Limitations with ECMP

When using flood reflectors, it is possible to encounter some issues with ECMP resulting in minimal usage of the paths through the network. The following cases represent some issues with ECMP:

- **ECMP Expected from Level 2 SPF not Realized in Forwarding**

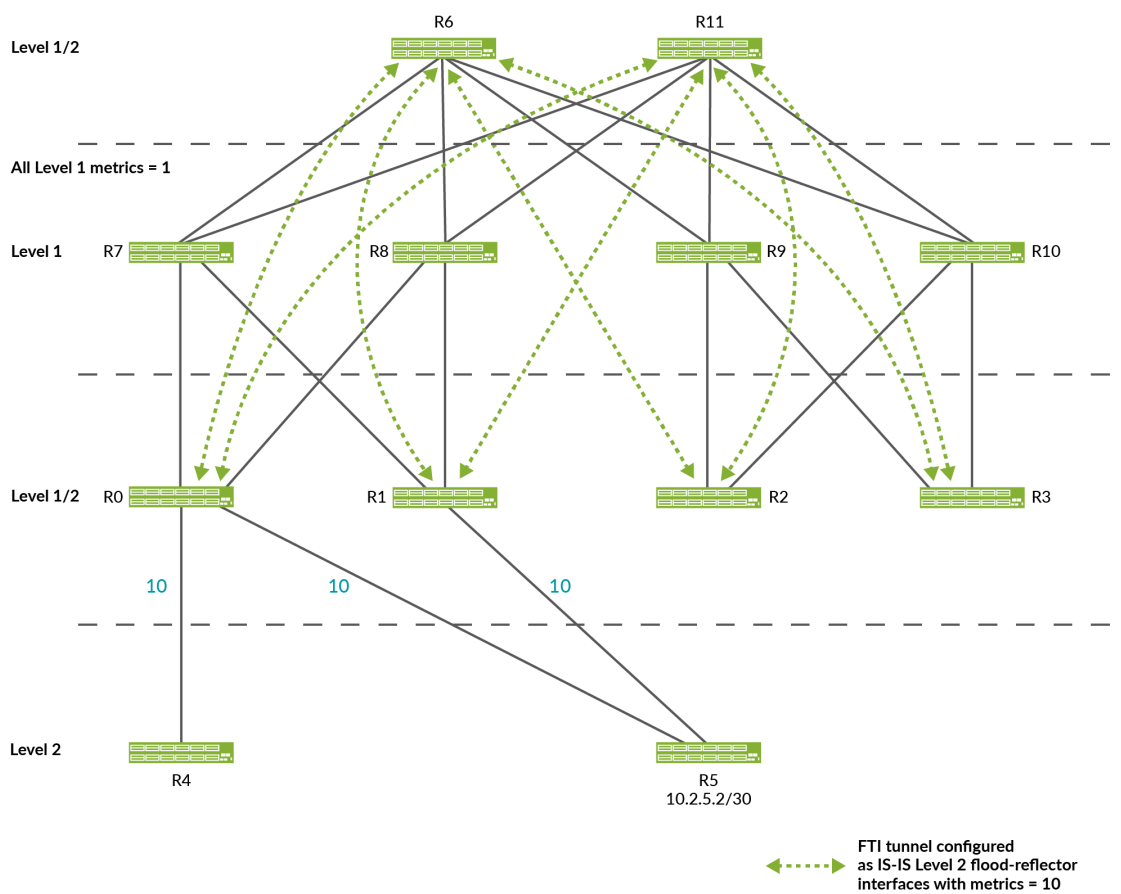


In this topology, based on its Level 2 SPF computation, R4 expects the traffic to enter the Level 1 fabric at R0 and exit at R1 and R2, with traffic being load balanced equally across the links from R1 to R5 and R2 to R5.

However, the Level 1 cost from R0 to R1 is 2 while the Level 1 cost from R0 to R2 is 4. All of the traffic is forwarded to R1 in the Level 1 fabric and uses the link from R1 to R5.

This issue can be addressed by building a full mesh of equal-cost Level 1 tunnels between each of the Level 1/ Level 2 leaf routers.

- **ECMP Expected from Level 2 SPF not Effectively Utilized**



Based on its Level 2 SPF computation, R4 expects the traffic to be sent only across the link from R0 to R5. This is the observed forwarding behavior. However, this behavior limits to utilize ECMP effectively. Building a full-mesh of equal-cost Level 1 tunnels between all of the Level 2/Level 1 leaf routers does not solve this issue.

## SEE ALSO

*flood-reflector*

## Example: IS-IS Flood Reflector

### IN THIS SECTION

- [Requirements | 245](#)
- [Overview | 245](#)
- [Configuration | 247](#)
- [Verification | 280](#)

This example shows how to configure flood reflectors in an IS-IS network. Flood reflection mainly reduces the redundancy in flooding paths and enhances the efficiency of IS-IS updates in large fabric topologies.

### Requirements

This example uses the following hardware and software components:

- Eleven PTX Series routers.
- Junos OS Release 20.4R1 or later running on all devices.

### Overview

#### IN THIS SECTION

- [Topology | 246](#)

Starting in Junos OS Release 20.4R1, you can configure flood-reflector interfaces in an IS-IS network. Flood reflection enables you to create IS-IS topologies where Level 1 areas provide transit forwarding for Level 2 destinations within a Level 2 topology. This is accomplished by creating Level 2 flood-reflection adjacencies within each Level 1 area.

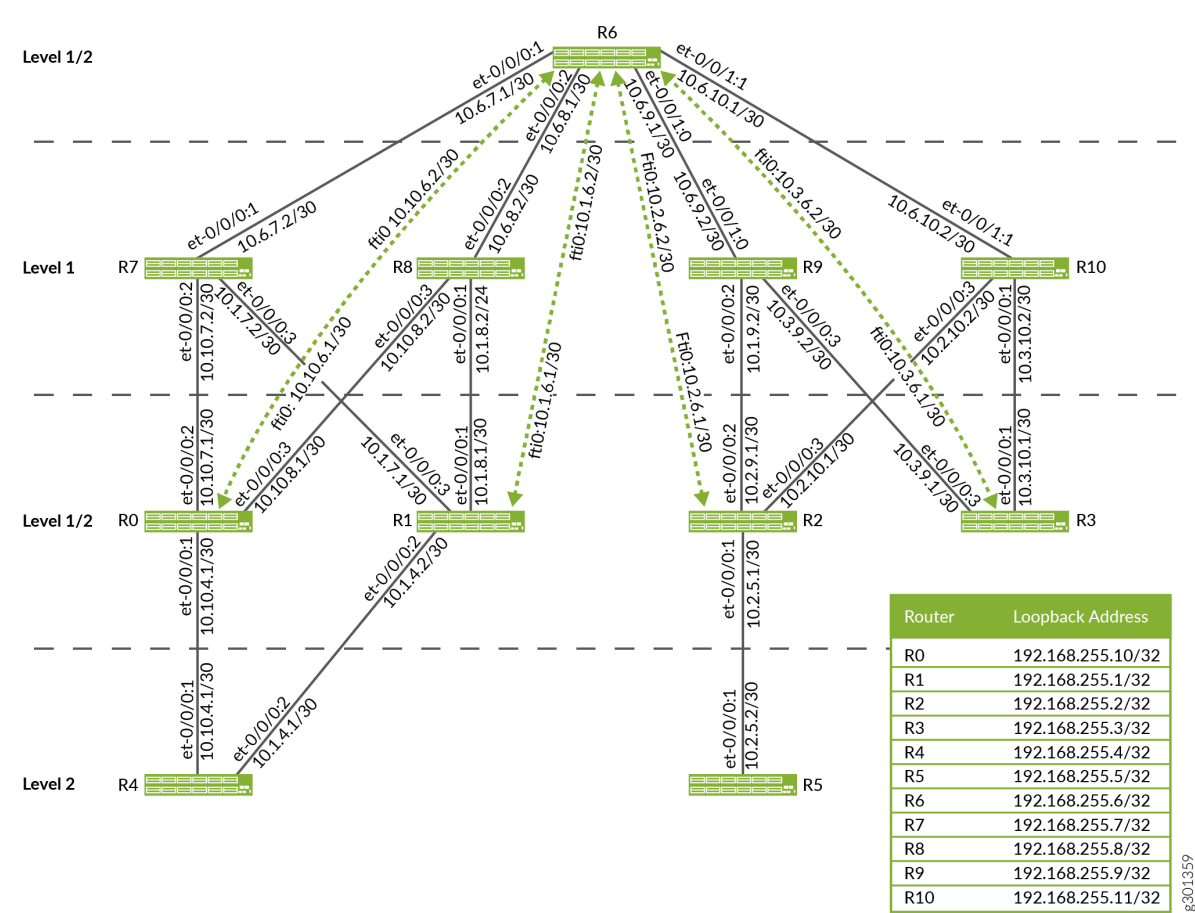
A flood-reflector adjacency reflects Level 2 link-state packet data units (PDUs) and they are used in the Level 2 shortest-path-first (SPF) computation. However, they are not used for forwarding.

To establish IS-IS adjacency for flood-reflection, flexible tunnel interfaces (FTI) are designated as flood-reflector interfaces. These tunnels utilize UDP encapsulation.

Topology

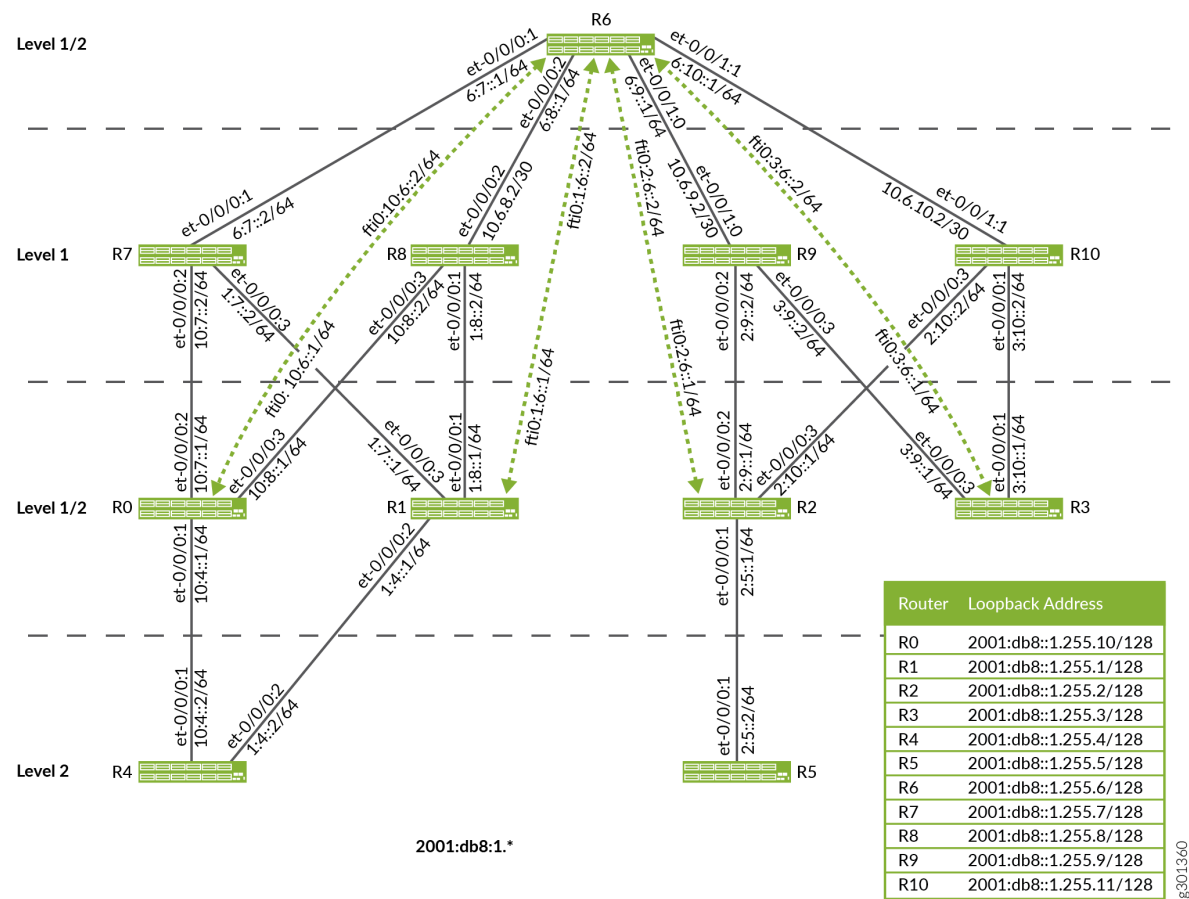
Figure 22 on page 246 shows a flood-reflector topology for IPv4 traffic and Figure 23 on page 247 shows a flood-reflector topology for IPv6 traffic in which Router R6 is the flood reflector. Routers R0, R1, R2, R3 are the flood-reflector clients that have FTI tunnels to R6. All of the FTI tunnels have metric 10 and are configured as flood-reflector interfaces. Routers R0, R1, R2, R3 are configured to redistribute Level 2 routes into Level 1 as Level 1/Level 2 inter-area routes. Routers R4 and R5 are Level 2 routers. Routers R7, R8, R9, R10 use the Level 1 route as they do not participate in flood reflection. The Level 1 metric is 100.

Figure 22: Flood-Reflector Topology for IPv4 Traffic



8301359

Figure 23: Flood-Reflector Topology for IPv6 addresses



## Configuration

### IN THIS SECTION

- CLI Quick Configuration | 248
- Configuring the Flood-Reflector Client | 257
- Configuring the Flood Reflector | 263
- Configuring the Non-Flood Reflector Device at Level 2 | 273
- Configuring the Non-Flood Reflector Device at Level 1 | 276

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter `commit` from the configuration mode.

#### Device R0

```
set interfaces et-0/0/0:1 description R0-to-R4
set interfaces et-0/0/0:1 unit 0 family inet address 10.10.4.1/30
set interfaces et-0/0/0:1 unit 0 family iso
set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:10:4::1/64
set interfaces et-0/0/0:2 description R0-to-R7
set interfaces et-0/0/0:2 unit 0 family inet address 10.10.7.1/30
set interfaces et-0/0/0:2 unit 0 family iso
set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:10:7::1/64
set interfaces et-0/0/0:3 description R0-to-R8
set interfaces et-0/0/0:3 unit 0 family inet address 10.10.8.1/30
set interfaces et-0/0/0:3 unit 0 family iso
set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:10:8::1/64
set interfaces lo0 unit 0 family inet address 192.168.255.10/32 primary
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5010.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::10/128
set routing-options router-id 192.168.255.10
set interfaces fti0 unit 0 tunnel encapsulation udp source address 192.168.255.10
set interfaces fti0 unit 0 tunnel encapsulation udp destination address 192.168.255.6
set interfaces fti0 unit 0 family inet address 10.10.6.1/30
set interfaces fti0 unit 0 family inet destination-udp-port 10000
set interfaces fti0 unit 0 family iso destination-udp-port 10030
set interfaces fti0 unit 0 family inet6 destination-udp-port 10010
set interfaces fti0 unit 0 family mpls destination-udp-port 10020
set protocols isis interface et-0/0/0:1.0 level 1 disable
set protocols isis interface et-0/0/0:1.0 level 2 metric 10
set protocols isis interface et-0/0/0:1.0 point-to-point
set protocols isis interface et-0/0/0:2.0 level 2 disable
set protocols isis interface et-0/0/0:2.0 level 1 metric 100
set protocols isis interface et-0/0/0:2.0 point-to-point
set protocols isis interface et-0/0/0:3.0 level 2 disable
set protocols isis interface et-0/0/0:3.0 level 1 metric 100
set protocols isis interface et-0/0/0:3.0 point-to-point
set protocols isis interface fti0.0 level 1 disable
set protocols isis interface fti0.0 level 2 flood-reflector cluster-id 100
```

```

set protocols isis interface lo0.0 passive
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only
set protocols isis level 2 flood-reflector client
set protocols isis export l2_l1_leak

```

## Device R1

```

set interfaces et-0/0/0:1 description R1-to-R8
set interfaces et-0/0/0:1 unit 0 family inet address 10.1.8.1/30
set interfaces et-0/0/0:1 unit 0 family iso
set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:1:7::1/64
set interfaces et-0/0/0:2 description R1-to-R4
set interfaces et-0/0/0:2 unit 0 family inet address 10.1.4.1/30
set interfaces et-0/0/0:2 unit 0 family iso
set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:1:8::1/64
set interfaces et-0/0/0:3 description R1-to-R7
set interfaces et-0/0/0:3 unit 0 family inet address 10.1.7.1/30
set interfaces et-0/0/0:3 unit 0 family iso
set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:1:4::1/64
set interfaces lo0 unit 0 family inet address 192.168.255.1/32 primary
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5001.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::1/128
set routing-options router-id 192.168.255.1
set interfaces fti0 unit 0 tunnel encapsulation udp source address 192.168.255.1
set interfaces fti0 unit 0 tunnel encapsulation udp destination address 192.168.255.6
set interfaces fti0 unit 0 family inet address 10.1.6.1/30
set interfaces fti0 unit 0 family inet destination-udp-port 10000
set interfaces fti0 unit 0 family iso destination-udp-port 10030
set interfaces fti0 unit 0 family inet6 destination-udp-port 10010
set interfaces fti0 unit 0 family mpls destination-udp-port 10020
set protocols isis interface et-0/0/0:1.0 level 2 disable
set protocols isis interface et-0/0/0:1.0 level 1 metric 100
set protocols isis interface et-0/0/0:1.0 point-to-point
set protocols isis interface et-0/0/0:2.0 level 1 disable
set protocols isis interface et-0/0/0:2.0 level 2 metric 11
set protocols isis interface et-0/0/0:2.0 point-to-point
set protocols isis interface et-0/0/0:3.0 level 2 disable
set protocols isis interface et-0/0/0:3.0 level 1 metric 100
set protocols isis interface et-0/0/0:3.0 point-to-point
set protocols isis interface fti0.0 level 1 disable
set protocols isis interface fti0.0 level 2 flood-reflector cluster-id 100

```

```

set protocols isis interface lo0.0 passive
set protocols isis level 2 wide-metrics-only
set protocols isis level 2 flood-reflector client
set protocols isis level 1 wide-metrics-only
set protocols isis export l2_l1_leak

```

## Device R2

```

set interfaces et-0/0/0:1 description R2-to-R5
set interfaces et-0/0/0:1 unit 0 family inet address 10.2.5.1/30
set interfaces et-0/0/0:1 unit 0 family iso
set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:2:5::1/64
set interfaces et-0/0/0:2 description R2-to-R9
set interfaces et-0/0/0:2 unit 0 family inet address 10.2.9.1/30
set interfaces et-0/0/0:2 unit 0 family iso
set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:2:9::1/64
set interfaces et-0/0/0:3 description R2-to-R10
set interfaces et-0/0/0:3 unit 0 family inet address 10.2.10.1/30
set interfaces et-0/0/0:3 unit 0 family iso
set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:2:10::1/64
set interfaces lo0 unit 0 family inet address 192.168.255.2/32 primary
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5002.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::2/128
set routing-options router-id 192.168.255.2
set interfaces fti0 unit 0 tunnel encapsulation udp source address 192.168.255.2
set interfaces fti0 unit 0 tunnel encapsulation udp destination address 192.168.255.6
set interfaces fti0 unit 0 family inet address 10.2.6.1/30
set interfaces fti0 unit 0 family inet destination-udp-port 10000
set interfaces fti0 unit 0 family iso destination-udp-port 10030
set interfaces fti0 unit 0 family inet6 destination-udp-port 10010
set interfaces fti0 unit 0 family mpls destination-udp-port 10020
set protocols isis interface et-0/0/0:1.0 level 1 disable
set protocols isis interface et-0/0/0:1.0 level 2 metric 10
set protocols isis interface et-0/0/0:1.0 point-to-point
set protocols isis interface et-0/0/0:2.0 level 2 disable
set protocols isis interface et-0/0/0:2.0 level 1 metric 100
set protocols isis interface et-0/0/0:2.0 point-to-point
set protocols isis interface et-0/0/0:3.0 level 2 disable
set protocols isis interface et-0/0/0:3.0 level 1 metric 100
set protocols isis interface et-0/0/0:3.0 point-to-point
set protocols isis interface fti0.0 level 1 disable
set protocols isis interface fti0.0 level 2 flood-reflector cluster-id 100

```



```

set protocols isis interface lo0.0 passive
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only
set protocols isis level 2 flood-reflector client
set protocols isis export l2_l1_leak

```

### Device R3

```

set interfaces et-0/0/0:1 description R3-to-R10
set interfaces et-0/0/0:1 unit 0 family inet address 10.3.10.1/30
set interfaces et-0/0/0:1 unit 0 family iso
set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:3:10::1/64
set interfaces et-0/0/0:3 description R3-to-R9
set interfaces et-0/0/0:3 unit 0 family inet address 10.3.9.1/30
set interfaces et-0/0/0:3 unit 0 family iso
set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:3:9::1/64
set interfaces fti0 unit 0 tunnel encapsulation udp source address 192.168.255.3
set interfaces fti0 unit 0 tunnel encapsulation udp destination address 192.168.255.6
set interfaces fti0 unit 0 family inet address 10.3.6.1/30
set interfaces fti0 unit 0 family inet destination-udp-port 10000
set interfaces fti0 unit 0 family iso destination-udp-port 10030
set interfaces fti0 unit 0 family inet6 destination-udp-port 10010
set interfaces fti0 unit 0 family mpls destination-udp-port 10020
set interfaces lo0 unit 0 family inet address 192.168.255.3/32 primary
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5003.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::3/128
set routing-options router-id 192.168.255.3
set protocols isis interface et-0/0/0:1.0 level 2 disable
set protocols isis interface et-0/0/0:1.0 level 1 metric 100
set protocols isis interface et-0/0/0:1.0 point-to-point
set protocols isis interface et-0/0/0:3.0 level 2 disable
set protocols isis interface et-0/0/0:3.0 level 1 metric 100
set protocols isis interface et-0/0/0:3.0 point-to-point
set protocols isis interface fti0.0 level 1 disable
set protocols isis interface fti0.0 level 2 flood-reflector cluster-id 100
set protocols isis interface lo0.0 passive
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only
set protocols isis level 2 flood-reflector client
set protocols isis export l2_l1_leak

```

## Device R4

```

set interfaces et-0/0/0:1 description R4-to-R0
set interfaces et-0/0/0:1 unit 0 family inet address 10.10.4.2/30
set interfaces et-0/0/0:1 unit 0 family iso
  set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:10:4::2/64
set interfaces et-0/0/0:2 description R4-to-R1
set interfaces et-0/0/0:2 unit 0 family inet address 10.1.4.2/30
set interfaces et-0/0/0:2 unit 0 family iso
set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:1:4::2/64
set interfaces lo0 unit 0 family inet address 192.168.255.4/32 primary
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5004.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::4/128
set routing-options router-id 192.168.255.4
set protocols isis interface et-0/0/0:1.0 level 1 disable
set protocols isis interface et-0/0/0:1.0 level 2 metric 10
set protocols isis interface et-0/0/0:1.0 point-to-point
set protocols isis interface et-0/0/0:2.0 level 1 disable
set protocols isis interface et-0/0/0:2.0 level 2 metric 11
set protocols isis interface et-0/0/0:2.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only

```

## Device R5

```

set interfaces et-0/0/0:1 description R5-to-R2
set interfaces et-0/0/0:1 unit 0 family inet address 10.2.5.2/30
set interfaces et-0/0/0:1 unit 0 family iso
set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:2:5::2/64
set interfaces lo0 unit 0 family inet address 192.168.255.5/30 primary
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5005.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::5/128
set routing-options router-id 192.168.255.5
set protocols isis interface et-0/0/0:1.0 level 1 disable
set protocols isis interface et-0/0/0:1.0 level 2 metric 10
set protocols isis interface et-0/0/0:1.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only

```

## Device R6

```

set interfaces et-0/0/0:1 description R6-to-R7
set interfaces et-0/0/0:1 unit 0 family inet address 10.6.7.1/30
set interfaces et-0/0/0:1 unit 0 family iso
set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:6:7::1/64
set interfaces et-0/0/0:2 description R6-to-R8
set interfaces et-0/0/0:2 unit 0 family inet address 10.6.8.1/30
set interfaces et-0/0/0:2 unit 0 family iso
set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:6:8::1/64
set interfaces et-0/0/1:0 description R6-to-R9-Link
set interfaces et-0/0/1:0 unit 0 family inet address 10.6.9.1/30
set interfaces et-0/0/1:0 unit 0 family iso
set interfaces et-0/0/1:0 unit 0 family inet6 address 2001:db8:1:6:9::1/64
set interfaces et-0/0/1:1 description R6-to-R10
set interfaces et-0/0/1:1 unit 0 family inet address 10.6.10.1/30
set interfaces et-0/0/1:1 unit 0 family iso
set interfaces et-0/0/1:1 unit 0 family inet6 address 2001:db8:1:6:10::1/64
set interfaces lo0 unit 0 family inet address 192.168.255.6/32 primary
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5006.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::6/128
set routing-options router-id 192.168.255.6
set interfaces fti0 unit 0 tunnel encapsulation udp source address 192.168.255.6
set interfaces fti0 unit 0 tunnel encapsulation udp destination address 192.168.255.10
set interfaces fti0 unit 0 family inet address 10.10.6.2/30
set interfaces fti0 unit 0 family inet destination-udp-port 10000
set interfaces fti0 unit 0 family iso destination-udp-port 10030
set interfaces fti0 unit 0 family inet6 destination-udp-port 10010
set interfaces fti0 unit 0 family mpls destination-udp-port 10020
set interfaces fti0 unit 1 tunnel encapsulation udp source address 192.168.255.6
set interfaces fti0 unit 1 tunnel encapsulation udp destination address 192.168.255.1
set interfaces fti0 unit 1 family inet address 10.1.6.2/30
set interfaces fti0 unit 1 family inet destination-udp-port 10000
set interfaces fti0 unit 1 family iso destination-udp-port 10030
set interfaces fti0 unit 1 family inet6 destination-udp-port 10010
set interfaces fti0 unit 1 family mpls destination-udp-port 10020
set interfaces fti0 unit 2 tunnel encapsulation udp source address 192.168.255.6
set interfaces fti0 unit 2 tunnel encapsulation udp destination address 192.168.255.2
set interfaces fti0 unit 2 family inet address 10.2.6.2/30
set interfaces fti0 unit 2 family inet destination-udp-port 10000
set interfaces fti0 unit 2 family iso destination-udp-port 10030
set interfaces fti0 unit 2 family inet6 destination-udp-port 10010

```

```

set interfaces fti0 unit 2 family mpls destination-udp-port 10020
set interfaces fti0 unit 3 tunnel encapsulation udp source address 192.168.255.6
set interfaces fti0 unit 3 tunnel encapsulation udp destination address 192.168.255.3
set interfaces fti0 unit 3 family inet address 10.3.6.2/30
set interfaces fti0 unit 3 family inet destination-udp-port 10000
set interfaces fti0 unit 3 family iso destination-udp-port 10030
set interfaces fti0 unit 3 family inet6 destination-udp-port 10010
set interfaces fti0 unit 3 family mpls destination-udp-port 10020
set protocols isis interface et-0/0/0:1.0 level 2 disable
set protocols isis interface et-0/0/0:2.0 level 1 metric 100
set protocols isis interface et-0/0/0:2.0 level 2 disable
set protocols isis interface et-0/0/0:2.0 point-to-point
set protocols isis interface et-0/0/1:0.0 level 2 disable
set protocols isis interface et-0/0/1:0.0 level 1 metric 100
set protocols isis interface et-0/0/1:0.0 point-to-point
set protocols isis interface et-0/0/1:1.0 level 2 disable
set protocols isis interface et-0/0/1:1.0 level 1 metric 100
set protocols isis interface et-0/0/1:1.0 point-to-point
set protocols isis interface fti0.0 level 1 disable
set protocols isis interface fti0.0 level 2 flood-reflector
set protocols isis interface fti0.1 level 1 disable
set protocols isis interface fti0.1 level 2 flood-reflector
set protocols isis interface fti0.2 level 1 disable
set protocols isis interface fti0.2 level 2 flood-reflector
set protocols isis interface fti0.3 level 1 disable
set protocols isis interface fti0.3 level 2 flood-reflector
set protocols isis interface lo0.0 level 2 disable
set protocols isis interface lo0.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only
set protocols isis level 2 flood-reflector reflector cluster-id 100
set protocols isis export l2_l1_leak

```

## Device R7

```

set interfaces et-0/0/0:1 description R7-to-R6
set interfaces et-0/0/0:1 unit 0 family inet address 10.6.7.2/30
set interfaces et-0/0/0:1 unit 0 family iso
set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:6:7::2/64
set interfaces et-0/0/0:2 description R7-to-R0
set interfaces et-0/0/0:2 unit 0 family inet address 10.10.7.2/30

```

```

set interfaces et-0/0/0:2 unit 0 family iso
set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:10:7::2/64
set interfaces et-0/0/0:3 description R7-to-R1
set interfaces et-0/0/0:3 unit 0 family inet address 10.1.7.2/30
set interfaces et-0/0/0:3 unit 0 family iso
set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:1:7::2/64
set interfaces lo0 unit 0 family inet address 192.168.255.7/32 primary
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5007.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::7/128
set routing-options router-id 192.168.255.7
set protocols isis interface et-0/0/0:1.0 level 2 disable
set protocols isis interface et-0/0/0:1.0 level 1 metric 100
set protocols isis interface et-0/0/0:1.0 point-to-point
set protocols isis interface et-0/0/0:2.0 level 2 disable
set protocols isis interface et-0/0/0:2.0 level 1 metric 100
set protocols isis interface et-0/0/0:2.0 point-to-point
set protocols isis interface et-0/0/0:3.0 level 2 disable
set protocols isis interface et-0/0/0:3.0 level 1 metric 100
set protocols isis interface et-0/0/0:3.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only

```

## Device R8

```

set interfaces et-0/0/0:1 description R8-to-R1
set interfaces et-0/0/0:1 unit 0 family inet address 10.1.8.2/30
set interfaces et-0/0/0:1 unit 0 family iso
set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:1:8::2/64
set interfaces et-0/0/0:2 description R8-to-R6
set interfaces et-0/0/0:2 unit 0 family inet address 10.6.8.2/30
set interfaces et-0/0/0:2 unit 0 family iso
set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:6:8::2/64
set interfaces et-0/0/0:3 description R8-to-R0
set interfaces et-0/0/0:3 unit 0 family inet address 10.10.8.2/30
set interfaces et-0/0/0:3 unit 0 family iso
set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:10:8::2/64
set interfaces lo0 unit 0 family inet address 192.168.255.11/32 primary
set interfaces lo0 unit 0 family inet address 192.168.255.8/32 primary
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::8/128
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5008.00
set routing-options router-id 192.168.255.11

```

```

set protocols isis interface et-0/0/0:1.0 level 2 disable
set protocols isis interface et-0/0/0:1.0 level 1 metric 100
set protocols isis interface et-0/0/0:1.0 point-to-point
set protocols isis interface et-0/0/0:2.0 level 2 disable
set protocols isis interface et-0/0/0:2.0 level 1 metric 100
set protocols isis interface et-0/0/0:2.0 point-to-point
set protocols isis interface et-0/0/0:3.0 level 2 disable
set protocols isis interface et-0/0/0:3.0 level 1 metric 100
set protocols isis interface et-0/0/0:3.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only

```

## Device R9

```

set interfaces et-0/0/0:2 description R9-to-R2
set interfaces et-0/0/0:2 unit 0 family inet address 10.2.9.2/30
set interfaces et-0/0/0:2 unit 0 family iso
set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:2:9::2/64
set interfaces et-0/0/0:3 description R9-to-R3
set interfaces et-0/0/0:3 unit 0 family inet address 10.3.9.2/30
set interfaces et-0/0/0:3 unit 0 family iso
set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:3:9::2/64
set interfaces et-0/0/1:0 description R9-to-R6
set interfaces et-0/0/1:0 unit 0 family inet address 10.6.9.2/30
set interfaces et-0/0/1:0 unit 0 family iso
set interfaces et-0/0/1:0 unit 0 family inet6 address 2001:db8:1:6:9::2/64
set interfaces lo0 unit 0 family inet address 192.168.255.9/32 primary
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::9/128
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5009.00
set routing-options router-id 192.168.255.9
set protocols isis interface et-0/0/0:2.0 level 2 disable
set protocols isis interface et-0/0/0:2.0 level 1 metric 100
set protocols isis interface et-0/0/0:2.0 point-to-point
set protocols isis interface et-0/0/0:3.0 level 2 disable
set protocols isis interface et-0/0/0:3.0 level 1 metric 100
set protocols isis interface et-0/0/0:3.0 point-to-point
set protocols isis interface et-0/0/1:0.0 level 2 disable
set protocols isis interface et-0/0/1:0.0 level 1 metric 100
set protocols isis interface et-0/0/1:0.0 point-to-point
set protocols isis interface lo0.0 passive

```

```
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only
```

## Device R10

```
set interfaces et-0/0/0:1 description R10-to-R3
set interfaces et-0/0/0:1 unit 0 family inet address 10.3.10.2/30
set interfaces et-0/0/0:1 unit 0 family iso
set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:3:10::2/64
set interfaces et-0/0/0:3 description R10-to-R2
set interfaces et-0/0/0:3 unit 0 family inet address 10.2.10.2/30
set interfaces et-0/0/0:3 unit 0 family iso
set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:2:10::2/64
set interfaces et-0/0/1:1 description R10-to-R6
set interfaces et-0/0/1:1 unit 0 family inet address 10.6.10.2/30
set interfaces et-0/0/1:1 unit 0 family iso
set interfaces et-0/0/1:1 unit 0 family inet6 address 2001:db8:1:6:10::2/64
set interfaces lo0 unit 0 family inet address 192.168.255.11/32 primary
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::11/128
set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5011.00
set routing-options router-id 192.168.255.11
set protocols isis interface et-0/0/0:1.0 level 2 disable
set protocols isis interface et-0/0/0:1.0 level 1 metric 100
set protocols isis interface et-0/0/0:1.0 point-to-point
set protocols isis interface et-0/0/0:3.0 level 2 disable
set protocols isis interface et-0/0/0:3.0 level 1 metric 100
set protocols isis interface et-0/0/0:3.0 point-to-point
set protocols isis interface et-0/0/1:1.0 level 2 disable
set protocols isis interface et-0/0/1:1.0 level 1 metric 100
set protocols isis interface et-0/0/1:1.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis level 1 wide-metrics-only
set protocols isis level 2 wide-metrics-only
```

## *Configuring the Flood-Reflector Client*

### Step-by-Step Procedure

To configure the flood-reflector client R0, perform these tasks:

1. Configure the device interfaces to enable IP and ISO transport.

```
[edit]
user@R0#set interfaces et-0/0/0:1 unit 0 description "Connection Between R_0 and R_4"
user@R0#set interfaces et-0/0/0:1 unit 0 family inet address 10.10.4.1/30
user@R0#set interfaces et-0/0/0:1 unit 0 family iso
user@R0#set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:10:4::1/64
user@R0#set interfaces et-0/0/0:2 unit 0 description "Connection Between R_0 and R_7"
user@R0#set interfaces et-0/0/0:2 unit 0 family inet address 10.10.7.1/30
user@R0#set interfaces et-0/0/0:2 unit 0 family iso
user@R0#set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:10:7::1/64
user@R0#set interfaces et-0/0/0:3 unit 0 description "Connection Between R_0 and R_8"
user@R0#set interfaces et-0/0/0:3 unit 0 family inet address 10.10.8.1/30
user@R0#set interfaces et-0/0/0:3 unit 0 family iso
user@R0#set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:10:8::1/64
```

2. Configure the loopback interface (lo0) with IPv4 and IPv6 addresses that are used as router ID for IS-IS sessions. Configure an ISO network entity title (NET) address on the loopback interface for the router to support IS-IS.

```
[edit]
user@R0#set interfaces lo0 unit 0 family inet address 192.168.255.10/32 primary
user@R0#set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5010.00
user@R0#set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::10/128
```

3. Configure routing options to identify the router in the domain.

```
[edit]
user@R0#set routing-options router-id 192.168.255.10
```

4. Configure the source address for the FTI with UDP encapsulation. The source address is the router ID of the tunnel source.

```
[edit]
user@R0#set interfaces fti0 unit 0 tunnel encapsulation udp source address 192.168.255.10
```



5. Configure the destination address for the FTI. The destination address is the router ID of the tunnel endpoint.

```
[edit]
user@R0#set interfaces fti0 unit 0 tunnel encapsulation udp destination address
192.168.255.6
```

6. Specify the source IP address of the tunnel and the UDP port value of the destination that identifies the tunnel endpoint. Specify the value of destination-udp-port from 1 through 65,535.

```
[edit]
user@R0#set interfaces fti0 unit 0 family inet address 10.10.6.1/30
user@R0#set interfaces fti0 unit 0 family inet destination-udp-port 10000
user@R0#set interfaces fti0 unit 0 family iso destination-udp-port 10030
user@R0#set interfaces fti0 unit 0 family inet6 destination-udp-port 10010
user@R0#set interfaces fti0 unit 0 family mpls destination-udp-port 10020
```

7. Disable IS-IS Level 1 on the interface that connects to an interface of R4 at Level 2. Configure a metric of 10 for the Level 2 interface.

```
[edit]
user@R0#set protocols isis interface et-0/0/0:1.0 level 1 disable
user@R0#set protocols isis interface et-0/0/0:1.0 level 2 metric 10
```

8. Disable IS-IS Level 2 on the interface that connects to interfaces of R7 and R8 at Level 1. Configure a metric of 100 for the Level 1 interfaces.

```
[edit]
user@R0#set protocols isis interface et-0/0/0:2.0 level 1 metric 100
user@R0#set protocols isis interface et-0/0/0:2.0 level 2 disable
user@R0#set protocols isis interface et-0/0/0:3.0 level 1 metric 100
user@R0#set protocols isis interface et-0/0/0:3.0 level 2 disable
```

9. Configure the IS-IS interfaces to behave like point-to-point interfaces.

```
[edit]
user@R0#set protocols isis interface et-0/0/0:1.0 point-to-point
```

```

user@R0#set protocols isis interface et-0/0/0:2.0 point-to-point
user@R0#set protocols isis interface et-0/0/0:3.0 point-to-point

```

10. Disable Level 1 on the FTI and configure the flood reflector client at Level 2 by including the cluster identifier.

```

[edit]
user@R0#set protocols isis interface fti0.0 level 1 disable
user@R0#set protocols isis interface fti0.0 level 2 flood-reflector cluster-id 100
user@R0#set protocols isis level 2 flood-reflector client

```

11. Configure wide-metrics for IS-IS to allow wider range of metric values.

```

[edit]
user@R0#set protocols isis level 1 wide-metrics-only
user@R0#set protocols isis level 2 wide-metrics-only

```

12. Configure the loopback interface as passive to ensure the protocols do not run over the loopback interface and that the loopback interface is advertised correctly throughout the network.

```

[edit]
user@R0#set protocols isis interface lo0.0 passive

```

13. Enter commit from the configure mode on R0 device.

In addition to configuring these steps on flood-reflector client R0, repeat these steps for the flood-reflector clients R1, R2, R3 within the cluster that you configure.

## Results

From configuration mode, confirm your configuration by entering the, show interfaces, show routing-options, show protocols, and show policy-options commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```

interfaces {
  et-0/0/0:1 {
    unit 0 {
      family inet {
        address 10.10.4.1/30;

```

```

    }
    family iso;
    family inet6 {
        address 2001:db8:1:10:4::1/64;
    }
}
}
et-0/0/0:2 {
    unit 0 {
        family inet {
            address 10.10.7.1/30;
        }
        family iso;
        family inet6 {
            address 2001:db8:1:10:7::1/64;
        }
    }
}
et-0/0/0:3 {
    unit 0 {
        family inet {
            address 10.10.8.1/30;
        }
        family iso;
        family inet6 {
            address 2001:db8:1:10:8::1/64;
        }
    }
}
}
fti0 {
    unit 0 {
        tunnel {
            encapsulation udp {
                source {
                    address 192.168.255.10;
                }
                destination {
                    address 192.168.255.6;
                }
            }
        }
        family inet {
            address 10.10.6.1/30;
        }
    }
}

```

```

        destination-udp-port 10000;
    }
    family iso {
        destination-udp-port 10030;
    }
    family inet6 {
        destination-udp-port 10010;
    }
    family mpls {
        destination-udp-port 10020;
    }
}
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.255.10/32;
        }
        family iso {
            address 49.0001.1921.6825.5010.00;
        }
        family inet6 {
            address 2001:db8:1:255::10/128;
        }
    }
}
}
routing-options {
    router-id 192.168.255.10;
}
protocols {
    isis {
        interface et-0/0/0:1.0 {
            level 1 disable;
            level 2 {
                metric 10;
            }
            point-to-point;
        }
        interface et-0/0/0:2.0 {
            level 2 disable;
            level 1 {
                metric 100;
            }
        }
    }
}

```

```

    }
    point-to-point;
}
interface et-0/0/0:3.0;
    level 2 disable;
    level 1 {
        metric 100;
    }
    point-to-point;
}
interface fti0.0 {
    level 1 disable;
    level 2 {
        flood-reflector {
            cluster-id 100;
        }
    }
}
}
interface lo0.0 {
    passive;
}
}
level 1 wide-metrics-only;
level 2 {
    wide-metrics-only;
    flood-reflector {
        client;
    }
}
}
export [ l2_l1_leak ];

```

### *Configuring the Flood Reflector*

#### **Step-by-Step Procedure**

To configure the flood-reflector device R6, perform these tasks:

1. Configure the device interfaces to enable IP and ISO transport.

[edit]

```

user@R6#set interfaces et-0/0/0:1 unit 0 description "Connection Between R_6 and R_7"
user@R6#set interfaces et-0/0/0:1 unit 0 family inet address 10.6.7.1/30

```

```

user@R6#set interfaces et-0/0/0:1 unit 0 family iso
user@R6#set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:6:7::1/64
user@R6#set interfaces et-0/0/0:2 unit 0 description "Connection Between R_6 and R_8"
user@R6#set interfaces et-0/0/0:2 unit 0 family inet address 10.6.8.1/30
user@R6#set interfaces et-0/0/0:2 unit 0 family iso
user@R6#set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:6:8::1/64
user@R6#set interfaces et-0/0/0:3 unit 0 description "Connection Between R_6 and R_9"
user@R6#set interfaces et-0/0/0:3 unit 0 family inet address 10.6.9.1/30
user@R6#set interfaces et-0/0/0:3 unit 0 family iso
user@R6#set interfaces et-0/0/0:3 unit 0 family inet6 address 2001:db8:1:6:9::1/64
user@R6#set interfaces et-0/0/1:1 unit 0 description "Connection Between R_6 and R_10"
user@R6#set interfaces et-0/0/1:1 unit 0 family inet address 10.6.10.1/30
user@R6#set interfaces et-0/0/1:1 unit 0 family iso
user@R6#set interfaces et-0/0/1:1 unit 0 family inet6 address 2001:db8:1:6:10::1/64

```

2. Configure the loopback interface (lo0) with IPv4 and IPv6 addresses that are used as router ID for IS-IS sessions. Configure an ISO network entity title (NET) address on the loopback interface for the router to support IS-IS.

```

[edit]
user@R6#set interfaces lo0 unit 0 family inet address 192.168.255.6/32 primary
user@R6#set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5006.00
user@R6#set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::6/128

```

3. Configure routing options to identify the router in the domain.

```

[edit]
user@R6#set routing-options router-id 192.168.255.6

```

4. Configure the source address for the FTI with UDP encapsulation. The source address is the router ID of the tunnel source.

```

[edit]
user@R6#set interfaces fti0 unit 0 tunnel encapsulation udp source address 192.168.255.6

```

5. Configure the destination addresses for the FTI per unit. The destination addresses are the router IDs of the tunnel endpoints.

```
[edit]
user@R6#set interfaces fti0 unit 0 tunnel encapsulation udp destination address
192.168.255.10
user@R6#set interfaces fti0 unit 1 tunnel encapsulation udp destination address
192.168.255.1
user@R6#set interfaces fti0 unit 2 tunnel encapsulation udp destination address
192.168.255.2
user@R6#set interfaces fti0 unit 3 tunnel encapsulation udp destination address
192.168.255.3
```

6. Specify the source IP address of the tunnel and the UDP port value of the destination that identifies the tunnel endpoint per unit. Specify the value of destination-udp-port from 1 through 65,535.

```
[edit]
user@R6#set interfaces fti0 unit 0 family inet address 10.10.6.2/30
user@R6#set interfaces fti0 unit 0 family inet destination-udp-port 10000
user@R6#set interfaces fti0 unit 0 family iso destination-udp-port 10030
user@R6#set interfaces fti0 unit 0 family inet6 destination-udp-port 10010
user@R6#set interfaces fti0 unit 0 family mpls destination-udp-port 10020
user@R6#set interfaces fti0 unit 1 family inet address 10.1.6.2/30
user@R6#set interfaces fti0 unit 1 family inet destination-udp-port 10000
user@R6#set interfaces fti0 unit 1 family iso destination-udp-port 10030
user@R6#set interfaces fti0 unit 1 family inet6 destination-udp-port 10010
user@R6#set interfaces fti0 unit 1 family mpls destination-udp-port 10020
user@R6#set interfaces fti0 unit 2 family inet address 10.2.6.2/30
user@R6#set interfaces fti0 unit 2 family inet destination-udp-port 10000
user@R6#set interfaces fti0 unit 2 family inet6 destination-udp-port 10010
user@R6#set interfaces fti0 unit 2 family iso destination-udp-port 10030
user@R6#set interfaces fti0 unit 2 family mpls destination-udp-port 10020
user@R6#set interfaces fti0 unit 3 family inet address 10.3.6.2/30
user@R6#set interfaces fti0 unit 3 family inet destination-udp-port 10000
user@R6#set interfaces fti0 unit 3 family iso destination-udp-port 10030
user@R6#set interfaces fti0 unit 3 family inet6 destination-udp-port 10010
user@R6#set interfaces fti0 unit 3 family mpls destination-udp-port 10020
```

7. Configure a policy in the Level 2 area to leak routes into the Level 1 area.

```
[edit]
user@R6#set protocols isis export l2_l1_leak
```

8. Disable IS-IS Level 2 on the interface and configure a metric of 100 on the Level 1 interface.

```
[edit]
user@R6#set protocols isis interface et-0/0/0:1.0 level 2 disable
user@R6#set protocols isis interface et-0/0/0:1.0 level 1 metric 100
user@R6#set protocols isis interface et-0/0/0:2.0 level 2 disable
user@R6#set protocols isis interface et-0/0/0:2.0 level 1 metric 100
user@R6#set protocols isis interface et-0/0/1:0.0 level 2 disable
user@R6#set protocols isis interface et-0/0/1:0.0 level 1 metric 100
user@R6#set protocols isis interface et-0/0/1:1.0 level 2 disable
user@R6#set protocols isis interface et-0/0/1:1.0 level 1 metric 100
```

9. Configure the IS-IS interfaces to behave like point-to-point interfaces.

```
[edit]
user@R6#set protocols isis interface et-0/0/0:1.0 point-to-point
user@R6#set protocols isis interface et-0/0/0:2.0 point-to-point
user@R6#set protocols isis interface et-0/0/1:0.0 point-to-point
user@R6#set protocols isis interface et-0/0/1:1.0 point-to-point
```

10. Disable Level 1 on the FTI and configure the flood-reflector client at Level 2 by including the cluster identifier.

```
[edit]
user@R6#set protocols isis interface fti0.0 level 1 disable
user@R6#set protocols isis interface fti0.0 level 2 flood-reflector cluster-id 100
user@R6#set protocols isis interface fti0.1 level 1 disable
user@R6#set protocols isis interface fti0.1 level 2 flood-reflector
user@R6#set protocols isis interface fti0.2 level 1 disable
user@R6#set protocols isis interface fti0.2 level 2 flood-reflector
user@R6#set protocols isis interface fti0.3 level 1 disable
user@R6#set protocols isis interface fti0.3 level 2 flood-reflector
user@R6#set protocols isis level 2 flood-reflector reflector cluster-id 100
```



11. Configure wide-metrics for IS-IS to allow wider range of metric values.

```
[edit]
user@R6#set protocols isis level 1 wide-metrics-only
user@R6#set protocols isis level 2 wide-metrics-only
```

12. Configure the loopback interface as passive to ensure the protocols do not run over the loopback interface and that the loopback interface is advertised correctly throughout the network.

```
[edit]
user@R6#set protocols isis interface lo0.0 passive
```

13. Enter commit from the configure mode on R6 device.

## Results

From configuration mode, confirm your configuration by entering the , show interfaces, show routing-options, show protocols, and show policy-options commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
interfaces {
  et-0/0/0:1 {
    unit 0 {
      family inet {
        address 10.6.7.1/30;
      }
      family iso;
      family inet6 {
        address 2001:db8:1:6:7::1/64;
      }
    }
  }
  et-0/0/0:2 {
    unit 0 {
      family inet {
        address 10.6.8.1/30;
      }
      family iso;
      family inet6 {
        address 2001:db8:1:6:8::1/64;
      }
    }
  }
}
```

```

    }
  }
}
et-0/0/1:0 {
  unit 0 {
    family inet {
      address 10.6.9.1/30;
    }
    family iso;
    family inet6 {
      address 2001:db8:1:6:9::1/64;
    }
  }
}
et-0/0/1:1{
  unit 0 {
    family inet {
      address 10.6.10.1/30;
    }
    family iso;
    family inet6 {
      address 2001:db8:1:6:10::1/64;
    }
  }
}
fti0 {
  unit 0 {
    tunnel {
      encapsulation udp {
        source {
          address 192.168.255.6;
        }
        destination {
          address 192.168.255.10;
        }
      }
    }
    family inet {
      address 10.10.6.2/30;
      destination-udp-port 10000;
    }
    family iso {
      destination-udp-port 10030;
    }
  }
}

```

```

        }
        family inet6 {
            destination-udp-port 10010;
        }
        family mpls {
            destination-udp-port 10020;
        }
    }
}
fti0 {
    unit 1 {
        tunnel {
            encapsulation udp {
                source {
                    address 192.168.255.6;
                }
                destination {
                    address 192.168.255.1;
                }
            }
        }
        family inet {
            address 10.1.6.2/30;
            destination-udp-port 10000;
        }
        family iso {
            destination-udp-port 10030;
        }
        family inet6 {
            destination-udp-port 10010;
        }
        family mpls {
            destination-udp-port 10020;
        }
    }
}
fti0 {
    unit 2 {
        tunnel {
            encapsulation udp {
                source {
                    address 192.168.255.6;
                }
            }
        }
    }
}

```

```

        destination {
            address 192.168.255.2;
        }
    }
}
family inet {
    address 10.2.6.2/30;
    destination-udp-port 10000;
}
    family iso {
        destination-udp-port 10030;
    }
    family inet6 {
        destination-udp-port 10010;
    }
    family mpls {
        destination-udp-port 10020;
    }
}
}
fti0 {
    unit 3 {
        tunnel {
            encapsulation udp {
                source {
                    address 192.168.255.6;
                }
                destination {
                    address 192.168.255.3;
                }
            }
        }
    }
    family inet {
        address 10.3.6.2/30;
        destination-udp-port 10000;
    }
    family iso {
        destination-udp-port 10030;
    }
    family inet6 {
        destination-udp-port 10010;
    }
    family mpls {

```

```

        destination-udp-port 10020;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.255.6/32;
        }
        family iso {
            address 49.0001.1921.6825.5006.00;
        }
        family inet6 {
            address 2001:db8:1:255::6/128;
        }
    }
}
}
routing-options {
    router-id 192.168.255.6;
}
protocols {
    isis {
        interface et-0/0/0:1.0 {
            level 2 disable;
            level 1 {
                metric 100;
            }
            point-to-point;
        }
        interface et-0/0/0:2.0 {
            level 2 disable;
            level 1 {
                metric 100;
            }
            point-to-point;
        }
        interface et-0/0/1:0.0;
        level 2 disable;
        level 1 {
            metric 100;
        }
        point-to-point;
    }
}

```

```

    }
    interface et-0/0/1:1.0;
        level 2 disable;
        level 1 {
            metric 100;
        }
        point-to-point;
    }
    interface fti0.0 {
        level 1 disable;
        level 2 {
            flood-reflector {
                cluster-id 100;
            }
        }
    }
    interface fti0.1 {
        level 1 disable;
        level 2 {
            flood-reflector {
                cluster-id 100;
            }
        }
    }
    interface fti0.2 {
        level 1 disable;
        level 2 {
            flood-reflector {
                cluster-id 100;
            }
        }
    }
    interface fti0.3 {
        level 1 disable;
        level 2 {
            flood-reflector {
                cluster-id 100;
            }
        }
    }
    }
    interface lo0.0 {
        passive;
    }
}
level 1 wide-metrics-only;
level 2 {
    wide-metrics-only;

```

```

        flood-reflector {
            client:
        }
    }
    export [ l2_l1_leak ];
}

```

### *Configuring the Non-Flood Reflector Device at Level 2*

#### **Step-by-Step Procedure**

To configure the non-flood reflector device R4 at Level 2:

1. Configure the device interfaces to enable IP and ISO transport.

```

[edit]
user@R4#set interfaces et-0/0/0:1 unit 0 description "Connection Between R_4 and R_0"
user@R4#set interfaces et-0/0/0:1 unit 0 family inet address 10.10.4.2/30
user@R4#set interfaces et-0/0/0:1 unit 0 family iso
user@R4#set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:10:4::2/64
user@R4#set interfaces et-0/0/0:2 unit 0 description "Connection Between R_4 and R_1"
user@R4#set interfaces et-0/0/0:2 unit 0 family inet address 10.1.4.2/30
user@R4#set interfaces et-0/0/0:2 unit 0 family iso
user@R4#set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:1:4::2/64

```

2. Configure the loopback interface (lo0) with IPv4 and IPv6 addresses that are used as router ID for IS-IS sessions. Configure an ISO network entity title (NET) address on the loopback interface for the router to support IS-IS.

```

[edit]
user@R4#set interfaces lo0 unit 0 family inet address 192.168.255.4/32 primary
user@R4#set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5004.00
user@R4#set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::4/128

```

3. Configure routing options to identify the router in the domain.

```

[edit]
user@R4#set routing-options router-id 192.168.255.4

```

4. Disable IS-IS Level 1 on the interfaces and configure a metric of 10 and 11 on the Level 2 interfaces.

```
[edit]
user@R4#set protocols isis interface et-0/0/0:1.0 level 1 disable
user@R4#set protocols isis interface et-0/0/0:1.0 level 2 metric 10
user@R4#set protocols isis interface et-0/0/0:2.0 level 1 disable
user@R4#set protocols isis interface et-0/0/0:2.0 level 2 metric 11
```

5. Configure the IS-IS interfaces to behave like point-to-point interfaces.

```
[edit]
user@R4#set protocols isis interface et-0/0/0:1.0 point-to-point
user@R4#set protocols isis interface et-0/0/0:2.0 point-to-point
```

6. Configure wide-metrics for IS-IS to allow wider range of metric values.

```
[edit]
user@R4#set protocols isis level 1 wide-metrics-only
user@R4#set protocols isis level 2 wide-metrics-only
```

7. Configure the loopback interface as passive to ensure the protocols do not run over the loopback interface and that the loopback interface is advertised correctly throughout the network.

```
[edit]
user@R4#set protocols isis interface lo0.0 passive
```

8. Enter commit from the configure mode on R4 device.

In addition to configuring these steps on the Level 2 non-flood reflector device R4, repeat these steps for the non-flood reflector device R5.



## Results

From configuration mode, confirm your configuration by entering the , show interfaces, show routing-options, show protocols, and show policy-options commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```

interfaces {
  et-0/0/0:1 {
    unit 0 {
      family inet {
        address 10.10.4.2/30;
      }
      family iso;
      family inet6 {
        address 2001:db8:1:10:4::2/64;
      }
    }
  }
  et-0/0/0:2 {
    unit 0 {
      family inet {
        address 10.1.4.2/30;
      }
      family iso;
      family inet6 {
        address 2001:db8:1:1:4::2/64;
      }
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 192.168.255.4/32;
      }
      family iso {
        address 49.0001.1921.6825.5004.00;
      }
      family inet6 {
        address 2001:db8:1:255::4/128;
      }
    }
  }
}

```

```

}
routing-options {
    router-id 192.168.255.4;
}
protocols {
    isis {
        interface et-0/0/0:1.0 {
            level 1 disable;
            level 2 {
                metric 10;
            }
            point-to-point;
        }
        interface et-0/0/0:2.0 {
            level 1 disable;
            level 2 {
                metric 11;
            }
            point-to-point;
        }
        interface lo0.0 {
            passive;
        }
        level 1 wide-metrics-only;
        level 2 wide-metrics-only;
    }
    export [ l2_l1_leak ];
}

```

### *Configuring the Non-Flood Reflector Device at Level 1*

#### **Step-by-Step Procedure**

To configure the non-flood reflector device R7 at Level 1:

1. Configure the device interfaces to enable IP and ISO transport.

```

[edit]
user@R7#set interfaces et-0/0/0:1 unit 0 description "Connection Between R_4 and R_0"
user@R7#set interfaces et-0/0/0:1 unit 0 family inet address 10.10.4.2/30
user@R7#set interfaces et-0/0/0:1 unit 0 family iso

```

```

user@R7#set interfaces et-0/0/0:1 unit 0 family inet6 address 2001:db8:1:10:4::2/64
user@R7#set interfaces et-0/0/0:2 unit 0 description "Connection Between R_4 and R_1"
user@R7#set interfaces et-0/0/0:2 unit 0 family inet address 10.1.4.2/30
user@R7#set interfaces et-0/0/0:2 unit 0 family iso
user@R7#set interfaces et-0/0/0:2 unit 0 family inet6 address 2001:db8:1:1:4::2/64

```

2. Configure the loopback interface (lo0) with IPv4 and IPv6 addresses that are used as router ID for IS-IS sessions. Configure an ISO network entity title (NET) address on the loopback interface for the router to support IS-IS.

```

[edit]
user@R7#set interfaces lo0 unit 0 family inet address 192.168.255.4/32 primary
user@R7#set interfaces lo0 unit 0 family iso address 49.0001.1921.6825.5007.00
user@R7#set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::4/128

```

3. Configure routing options to identify the router in the domain.

```

[edit]
user@R7#set routing-options router-id 192.168.255.4

```

4. Disable Level 2 on the interfaces and configure a metric of 100 on the Level 1 interfaces.

```

[edit]
user@R7#set protocols isis interface et-0/0/0:1.0 level 2 disable
user@R7#set protocols isis interface et-0/0/0:1.0 level 1 metric 100
user@R7#set protocols isis interface et-0/0/0:2.0 level 2 disable
user@R7#set protocols isis interface et-0/0/0:3.0 level 2 disable
user@R7#set protocols isis interface et-0/0/0:3.0 level 1 metric 100

```

5. Configure the IS-IS interfaces to behave like point-to-point interfaces.

```

[edit]
user@R7#set protocols isis interface et-0/0/0:1.0 point-to-point
user@R7#set protocols isis interface et-0/0/0:2.0 point-to-point
user@R7#set protocols isis interface et-0/0/0:3.0 point-to-point

```

6. Configure wide-metrics for IS-IS to allow wider range of metric values.

```
[edit]
user@R7#set protocols isis level 1 wide-metrics-only
user@R7#set protocols isis level 2 wide-metrics-only
```

7. Configure the loopback interface as passive to ensure the protocols do not run over the loopback interface and that the loopback interface is advertised correctly throughout the network.

```
[edit]
user@R7#set protocols isis interface lo0.0 passive
```

8. Enter commit from the configure mode on R7 device.

In addition to configuring these steps on the Level 1 non-flood reflector device R7, repeat these steps for the non-flood reflector devices R8, R9, R10.

## Results

From configuration mode, confirm your configuration by entering the , show interfaces, show routing-options, show protocols, and show policy-options commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
interfaces {
  et-0/0/0:1 {
    unit 0 {
      family inet {
        address 10.6.7.2/30;
      }
      family iso;
      family inet6 {
        address 2001:db8:1:6:7::2/64;
      }
    }
  }
  et-0/0/0:2 {
    unit 0 {
      family inet {
        address 10.10.7.2/30;
      }
    }
  }
}
```

```

        family iso;
        family inet6 {
            address 2001:db8:1:10:7::2/64;
        }
    }
}
et-0/0/0:3 {
    unit 0 {
        family inet {
            address 10.1.7.2/30;
        }
        family iso;
        family inet6 {
            address 2001:db8:1:1:7::2/64;
        }
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.255.7/32;
        }
        family iso {
            address 49.0001.1921.6825.5007.00;
        }
        family inet6 {
            address 2001:db8:1:255::7/128;
        }
    }
}
}
routing-options {
    router-id 192.168.255.7;
}
protocols {
    isis {
        interface et-0/0/0:1.0 {
            level 2 disable;
            level 1 {
                metric 100;
            }
            point-to-point;
        }
    }
}

```

```

interface et-0/0/0:2.0 {
    level 2 disable;
    level 1 {
        metric 100;
    }
    point-to-point;
}
interface et-0/0/0:3.0 {
    level 2 disable;
    level 1 {
        metric 100;
    }
    point-to-point;
}
interface lo0.0 {
    passive;
}
level 1 wide-metrics-only;
level 2 wide-metrics-only;
export [ l2_l1_leak ];
}

```

## Verification

### IN THIS SECTION

- [Verify the IS-IS Adjacency | 281](#)
- [Verify the Flood Reflector Status | 282](#)
- [Verify Flood-Reflector Client Status | 283](#)
- [Verify the IS-IS Database | 283](#)
- [Verify the IS-IS Route | 285](#)

To confirm that the configuration is working properly, perform the following tasks:

## Verify the IS-IS Adjacency

### Purpose

Verify that the IS-IS adjacencies are up.

### Action

From operational mode, run the `show isis adjacency` command.

```
user@R6>show isis adjacency
```

R6

**Interface: fti0.0, Level: 2, State: Up, Expires in 21 secs**

Interface	System	L	State	Hold (secs)	SNPA
fti0.0	R0	2	Up	24	
fti0.1	R1	2	Up	21	
fti0.2	R2	2	Up	20	
fti0.3	R3	2	Up	20	
et-0/0/1:1.0	R10	1	Up	25	
et-0/0/0:1.0	R7	1	Up	22	
et-0/0/0:2.0	R8	1	Up	20	
et-0/0/1:0.0	R9	1	Up	20	

Verify that the IS-IS instance is running on devices R0 and R6 and that they are adjacent to each other.

From operational mode, run the `show isis adjacency extensive` command.

### On R0

```
user@R0>show isis R6 adjacency extensive
```

R6

**Interface: fti0.0, Level: 2, State: Up, Expires in 21 secs**

Priority: 0, Up/Down transitions: 1, Last transition: 13:48:14 ago

Circuit type: 2, Speaks: IP, IPv6

Topologies: Unicast

Restart capable: Yes, Adjacency advertisement: Advertise

IP addresses: 10.10.6.2

IPv6 addresses: fe80::5668:a30f:fc54:2a13

Level 2 Flood reflector client, Cluster-id: 100

Transition log:

When	State	Event	Down reason
Wed Nov 25 09:27:25	Up	Seenself	

## Meaning

- The interface fti0.0 on the device R0 has established adjacency with the device R6.

## Verify the Flood Reflector Status

## Purpose

Verify that the flood reflector is enabled and verify its status.

## Action

From operational mode, run the `show isis interface fti0.0 extensive` command.

## On R6

```
user@R6>show isis interface fti0.0 extensive

IS-IS interface database:
fti0.0
  Index: 76, State: 0x6, Circuit id: 0x1, Circuit type: 2
  LSP interval: 100 ms, CSNP interval: 40 s, Loose Hello padding, IIH max size: 1492
  Adjacency advertisement: Advertise, Layer2-map: Disabled
  Level 2 Flood reflector, Cluster-id: 100
  Interface Group Holddown Delay: 20 s, remaining: 0 s
  Level 1
    Adjacencies: 0, Priority: 64, Metric: 10
    Disabled
  Level 2
    Adjacencies: 1, Priority: 64, Metric: 10
    Hello Interval: 9.000 s, Hold Time: 27 s
```

## Meaning

- On R6, the flood reflector is enabled on fti0 at level 2 with cluster-id 100.



## *Verify Flood-Reflector Client Status*

### **Purpose**

Verify that the flood-reflector client is enabled and verify its status.

### **Action**

From operational mode, run the `show isis interface fti0.0 extensive` command.

### **On R0**

```
user@R0>show isis interface fti0.0 extensive

IS-IS interface database:
fti0.0
  Index: 76, State: 0x6, Circuit id: 0x1, Circuit type: 2
  LSP interval: 100 ms, CSNP interval: 20 s, Loose Hello padding, IIH max size: 1492
  Adjacency advertisement: Advertise, Layer2-map: Disabled
  Level 2 Flood reflector client, Cluster-id: 100
  Interface Group Holddown Delay: 20 s, remaining: 0 s
  Level 1
    Adjacencies: 0, Priority: 64, Metric: 10
    Disabled
  Level 2
    Adjacencies: 1, Priority: 64, Metric: 10
    Hello Interval: 9.000 s, Hold Time: 27 s
```

### **Meaning**

- On R0, the flood-reflector client is enabled on fti0 at level 2 with cluster-id 100.

## *Verify the IS-IS Database*

### **Purpose**

Verify the IS-IS database on the flood reflector and the flood-reflector clients.

### **Action**

From operational mode, run the `show isis database` command.

## On R0

```

user@R0>show isis database r0 extensive

IS-IS level 1 link-state database:

TLVs:
  Area address: 47.0005.80ff.f800.0000.0108.0001 (13)
  Area address: 49.0001 (3)
  LSP Buffer Size: 1492
  Speaks: IP
  Speaks: IPV6
  IP router id: 192.168.255.10
  IP address: 192.168.255.10
  IPv6 TE Router ID: abcd::128:53:69:151
  Hostname: r0
  Extended IS Reachability TLV, Type: 22, Length: 127
  IS extended neighbor: r4.00, Metric: default 10 SubTLV len: 58
    IP address: 10.10.4.1
    IPv6 address: 2001:db8:1:10:4::1
    Neighbor's IP address: 10.10.4.2
    Neighbor's IPv6 address: 2001:db8:1:10:4::2
    Local interface index: 73, Remote interface index: 73
  IS extended neighbor: R6.00, Metric: default 10 SubTLV len: 47
    IP address: 10.10.6.1
    IPv6 address: 70cc:ffff:bf0b:d0cb:bc78:e90d:70cc:ffff
    Neighbor's IP address: 10.10.6.2
    Local interface index: 76, Remote interface index: 77
Flood reflector client, Cluster-id: 100
  No queued transmissions

```

## On R6

```

user@R6>show isis database R6 extensive

Router Capability: Router ID 192.168.255.6, Flags: 0x00
  IPv6 TE Router Id: abcd::128:53:69:140
  Extended IS Reachability TLV, Type: 22, Length: 160
  IS extended neighbor: r0.00, Metric: default 10 SubTLV len: 29
    IP address: 10.10.6.2
    Neighbor's IP address: 10.10.6.1

```

```

Local interface index: 77, Remote interface index: 76
Flood reflector, Cluster-id: 100
IS extended neighbor: r1.00, Metric: default 10 SubTLV len: 29
IP address: 10.1.6.2
Neighbor's IP address: 10.1.6.1
Local interface index: 78, Remote interface index: 76
Flood reflector, Cluster-id: 100
IS extended neighbor: r3.00, Metric: default 10 SubTLV len: 29
IP address: 10.3.6.2
Neighbor's IP address: 10.3.6.1
Local interface index: 80, Remote interface index: 75
Flood reflector, Cluster-id: 100
IS extended neighbor: r2.00, Metric: default 10 SubTLV len: 29
IP address: 10.2.6.2
Neighbor's IP address: 10.2.6.1
Local interface index: 79, Remote interface index: 76
Flood reflector, Cluster-id: 100
No queued transmissions

```

## Meaning

On R0 and R6, the IS-IS database shows the flood-reflector client and the flood reflector with cluster-id 100.

## Verify the IS-IS Route

## Purpose

Verify that the Level 2 routes learned on the FTI are not installed in the R0 routing table.

## Action

From operational mode, run the `show isis route` command.

## On R0

```

user@R0>show isis route
10.1.4.0/30      2  220230      21 int  et-0/0/0:1.0  IPV4 r4
10.1.7.0/30      1  220222      200 int  et-0/0/0:2.0  IPV4 r7
10.1.8.0/30      1  220222      200 int  et-0/0/0:3.0  IPV4 r8
10.2.9.0/30      1  220222      400 int  et-0/0/0:3.0  IPV4 r8

```

10.2.10.0/30	1	220222	400	int	et-0/0/0:3.0	IPV4	r8
10.3.9.0/30	1	220222	400	int	et-0/0/0:3.0	IPV4	r8
10.3.10.0/30	1	220222	400	int	et-0/0/0:3.0	IPV4	r8
10.6.7.0/30	1	220222	200	int	et-0/0/0:2.0	IPV4	r7
10.6.8.0/30	1	220222	200	int	et-0/0/0:3.0	IPV4	r8
10.6.9.0/30	1	220222	300	int	et-0/0/0:3.0	IPV4	r8
10.6.10.0/30	1	220222	300	int	et-0/0/0:3.0	IPV4	r8
192.168.255.4/30	1	220222	410	int	et-0/0/0:3.0	IPV4	r8

## Meaning

The Level 2 routes learned on the FTI between R0 and R6 are not installed in the R0 routing table.

## SEE ALSO

| *flood-reflector*

# Configuring IS-IS Multitopology Routing and IPv6 Support

## IN THIS CHAPTER

- [IS-IS Multicast Topologies Overview | 287](#)
- [Example: Configuring IS-IS Multicast Topology | 289](#)
- [Understanding Dual Stacking of IPv4 and IPv6 Unicast Addresses | 311](#)
- [Example: Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses | 313](#)
- [Understanding IS-IS IPv4 and IPv6 Unicast Topologies | 322](#)
- [Example: Configuring IS-IS IPv4 and IPv6 Unicast Topologies | 323](#)

## IS-IS Multicast Topologies Overview

Most multicast routing protocols perform a reverse-path forwarding (RPF) check on the source of multicast data packets. If a packet comes in on the interface that is used to send data to the source, the packet is accepted and forwarded to one or more downstream interfaces. Otherwise, the packet is discarded and a notification is sent to the multicast routing protocol running on the interface.

In certain instances, the unicast routing table used for the RPF check is also the table used for forwarding unicast data packets. Thus, unicast and multicast routing are congruent. In other cases, where it is preferred that multicast routing be independent of unicast routing, the multicast routing protocols are configured to perform the RPF check using an alternate unicast routing table `inet.2`.

You can configure IS-IS to calculate an alternate IPv4 multicast topology, in addition to the normal IPv4 unicast topology, and add the corresponding routes to `inet.2`. The IS-IS interface metrics for the multicast topology can be configured independently of the unicast metrics. You can also selectively disable interfaces from participating in the multicast topology while continuing to participate in the regular unicast topology. This enables you to exercise control over the paths that multicast data takes through a network so that it is independent of unicast data paths. You can also configure IS-IS to calculate an alternate IPv6 multicast topology, in addition to the normal IPv6 unicast topology.



**NOTE:** IS-IS only starts advertising the routes when the interface routes are in inet.2.



**NOTE:** For the multicast metric commands, these are interface-specific settings, not global.

Table 3 on page 288 lists the various IPv4 statements you can use to configure IS-IS topologies.

**Table 3: IPv4 Statements**

Statement	Description
ipv4-multicast	Enables an alternate IPv4 multicast topology.
ipv4-multicast-metric <i>number</i>	Configures the multicast metric for an alternate IPv4 multicast topology.
no-ipv4-multicast	Excludes an interface from the IPv4 multicast topology.
no-unicast-topology	Excludes an interface from the IPv4 unicast topologies.

Table 4 on page 288 lists the various IPv6 statements you can use to configure IS-IS topologies.

**Table 4: IPv6 Statements**

Statement	Description
ipv6-multicast	Enables an alternate IPv6 multicast topology.
ipv6-unicast	Enables an alternate IPv6 unicast topology.
ipv6-multicast-metric <i>number</i>	Configures the multicast metric for an alternate IPv6 multicast topology.
ipv6-unicast-metric <i>number</i>	Configures the unicast metric for an alternate IPv6 multicast topology.

**Table 4: IPv6 Statements** *(Continued)*

Statement	Description
no-ipv6-multicast	Excludes an interface from the IPv6 multicast topology.
no-ipv6-unicast	Excludes an interface from the IPv6 unicast topologies.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

### RELATED DOCUMENTATION

| [Example: Configuring IS-IS Multicast Topology](#) | 289

## Example: Configuring IS-IS Multicast Topology

### IN THIS SECTION

- [Requirements](#) | 295
- [Overview](#) | 296
- [Verification](#) | 297

This example shows how to configure a multicast topology for an IS-IS network.

### Configuration

#### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

## Router R1

```

set protocols isis traceoptions file isis size 5m world-readable
set protocols isis traceoptions flag error
set protocols isis topologies ipv4-multicast
set protocols isis interface ge-0/0/0 level 1 metric 15
set protocols isis interface ge-0/0/0 level 1 ipv4-multicast-metric 18
set protocols isis interface ge-0/0/0 level 2 metric 20
set protocols isis interface ge-0/0/0 level 2 ipv4-multicast-metric 14
set protocols isis interface ge-1/0/0 level 1 metric 13
set protocols isis interface ge-1/0/0 level 1 ipv4-multicast-metric 12
set protocols isis interface ge-1/0/0 level 2 metric 29
set protocols isis interface ge-1/0/0 level 2 ipv4-multicast-metric 23
set protocols isis interface fxp0.0 disable

```

## Router R2

```

set protocols isis traceoptions file isis size 5m world-readable
set protocols isis traceoptions flag error
set protocols isis topologies ipv4-multicast
set protocols isis interface ge-0/0/0 level 1 metric 13
set protocols isis interface ge-0/0/0 level 1 ipv4-multicast-metric 12
set protocols isis interface ge-0/0/0 level 2 metric 29
set protocols isis interface ge-0/0/0 level 2 ipv4-multicast-metric 23
set protocols isis interface ge-1/0/0 level 1 metric 14
set protocols isis interface ge-1/0/0 level 1 ipv4-multicast-metric 18
set protocols isis interface ge-1/0/0 level 2 metric 32
set protocols isis interface ge-1/0/0 level 2 ipv4-multicast-metric 26
set protocols isis interface fxp0.0 disable

```

## Router R3

```

set protocols isis traceoptions file isis size 5m world-readable
set protocols isis traceoptions flag error
set protocols isis topologies ipv4-multicast
set protocols isis interface ge-0/0/0 level 1 metric 19
set protocols isis interface ge-0/0/0 level 1 ipv4-multicast-metric 11
set protocols isis interface ge-0/0/0 level 2 metric 27
set protocols isis interface ge-0/0/0 level 2 ipv4-multicast-metric 21
set protocols isis interface ge-1/0/0 level 1 metric 16
set protocols isis interface ge-1/0/0 level 1 ipv4-multicast-metric 26

```



```

set protocols isis interface ge-1/0/0 level 2 metric 30
set protocols isis interface ge-1/0/0 level 2 ipv4-multicast-metric 20
set protocols isis interface fxp0.0 disable

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure IS-IS multicast topologies:

1. Enable the multicast topology for IS-IS by using the `ipv4-multicast` statement.

Routers R1, R2, and R3

```

[edit protocols isis]
user@host# set traceoptions file isis size 5m world-readable
user@host# set traceoptions flag error
user@host# set topologies ipv4-multicast

```

2. Enable multicast metrics on the first Ethernet Interface by using the `ipv4-multicast-metric` statement.

Router R1

```

[edit protocols isis interface ge-0/0/0 ]
user@R1# set level 1 metric 15
user@R1# set level 1 ipv4-multicast-metric 18
user@R1# set level 2 metric 20
user@R1# set level 2 ipv4-multicast-metric 14

```

Router R2

```

[edit protocols isis interface ge-0/0/0]
user@R2# set level 1 metric 13
user@R2# set level 1 ipv4-multicast-metric 12
user@R2# set level 2 metric 29
user@R2# set level 2 ipv4-multicast-metric 23

```

## Router R3

```
[edit protocols isis interface ge-0/0/0]
user@R3# set level 1 metric 19
user@R3# set level 1 ipv4-multicast-metric 11
user@R3# set level 2 metric 27
user@R3# set level 2 ipv4-multicast-metric 21
```

3. Enable multicast metrics on a second ethernet Interface by using the `ipv4-multicast-metric` statement.

## Router R1

```
[edit protocols isis interface ge-1/0/0]
user@R1# set level 1 metric 13
user@R1# set level 1 ipv4-multicast-metric 12
user@R1# set level 2 metric 29
user@R1# set level 2 ipv4-multicast-metric 23
```

## Router R2

```
[edit protocols isis interface ge-1/0/0]
user@R2# set level 1 metric 14
user@R2# set level 1 ipv4-multicast-metric 18
user@R2# set level 2 metric 32
user@R2# set level 2 ipv4-multicast-metric 26
```

## Router R3

```
[edit protocols isis interface ge-1/0/0]
user@R3# set level 1 metric 16
user@R3# set level 1 ipv4-multicast-metric 26
user@R3# set level 2 metric 30
user@R3# set level 2 ipv4-multicast-metric 20
```

4. Disable the out-of-band management port, `fxp0`.

Routers R1, R2, and R3

```
[edit protocols isis]  
user@host# set interface fxp0.0 disable
```

5. If you are done configuring the routers, commit the configuration.

Routers R1, R2, and R3

```
[edit]  
user@host# commit
```

## Results

From configuration mode, confirm your configuration by using the `show protocols isis` statement. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

Router R1

```
user@R1# show protocols isis  
  
traceoptions {  
    file isis size 5m world-readable;  
    flag error;  
}  
topologies ipv4-multicast;  
interface ge-0/0/0 {  
    level 1 {  
        metric 15;  
        ipv4-multicast-metric 18;  
    }  
    level 2 {  
        metric 20;  
        ipv4-multicast-metric 14;  
    }  
}  
interface ge-1/0/0 {  
    level 1 {  
        metric 13;
```

```

        ipv4-multicast-metric 12;
    }
    level 2 {
        metric 29;
        ipv4-multicast-metric 23;
    }
}
interface fxp0.0 {
    disable;
}

```

## Router R2

```

user@R2# show protocols isis

traceoptions {
    file isis size 5m world-readable;
    flag error;
}
topologies ipv4-multicast;
interface ge-0/0/0 {
    level 1 {
        metric 13;
        ipv4-multicast-metric 12;
    }
    level 2 {
        metric 29;
        ipv4-multicast-metric 23;
    }
}
interface ge-1/0/0 {
    level 1 {
        metric 14;
        ipv4-multicast-metric 18;
    }
    level 2 {
        metric 32;
        ipv4-multicast-metric 26;
    }
}
interface fxp0.0 {

```

```

    disable;
}

```

## Router R3

```

user@R3# show protocols isis

traceoptions {
  file isis size 5m world-readable;
  flag error;
}
topologies ipv4-multicast;
interface ge-0/0/0 {
  level 1 {
    metric 19;
    ipv4-multicast-metric 11;
  }
  level 2 {
    metric 27;
    ipv4-multicast-metric 21;
  }
}
interface ge-1/0/0 {
  level 1 {
    metric 16;
    ipv4-multicast-metric 26;
  }
  level 2 {
    metric 30;
    ipv4-multicast-metric 20;
  }
}
interface fxp0.0 {
  disable;
}

```

## Requirements

Before you begin, configure IS-IS on all routers. See ["Example: Configuring IS-IS" on page 15](#) for information about the required IS-IS configuration.

This example uses the following hardware and software components:

- Three Routers
- Junps OS running on all devices

## Overview

### IN THIS SECTION

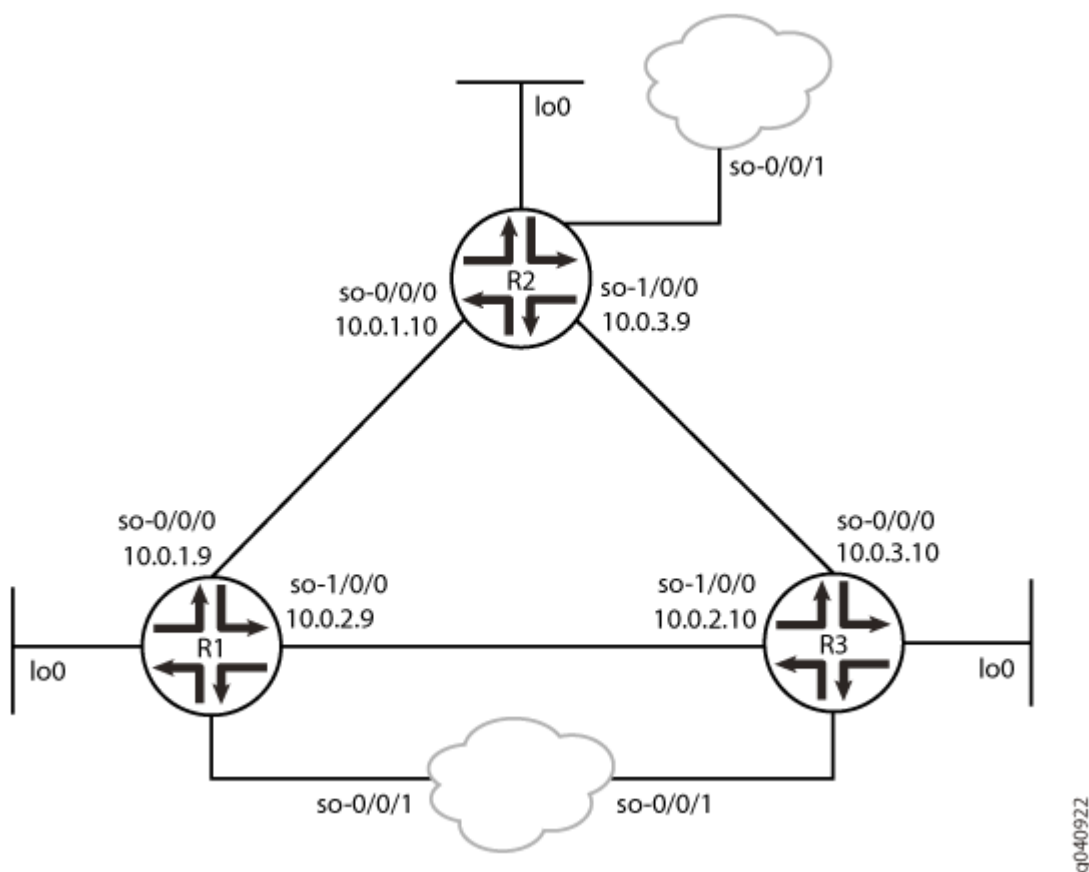
- [Topology | 296](#)

This example shows an IS-IS multicast topology configuration. Three routers are connected to each other. A loopback interface is configured on each router.

## Topology

[Figure 24 on page 297](#) shows the sample network.

Figure 24: Configuring IS-IS Multicast Topology



## Verification

### IN THIS SECTION

- [Verifying the Connection Between Routers R1, R2, and R3 | 298](#)
- [Verifying That IS-IS Is Configured | 300](#)
- [Verifying the Configured Multicast Metric Values | 303](#)
- [Verifying the Configuration of the Multicast Topology | 305](#)

Confirm that the configuration is working properly.

## Verifying the Connection Between Routers R1, R2, and R3

### Purpose

Make sure that Routers R1, R2, and R3 are connected to each other.

### Action

Ping the other two routers from any router, to check the connectivity between the three routers as per the network topology.

```
user@R1> ping 10.0.3.9
```

```
PING 10.0.3.9 (10.0.3.9): 56 data bytes
64 bytes from 10.0.3.9: icmp_seq=0 ttl=64 time=1.299 ms
64 bytes from 10.0.3.9: icmp_seq=1 ttl=64 time=52.304 ms
64 bytes from 10.0.3.9: icmp_seq=2 ttl=64 time=1.271 ms
64 bytes from 10.0.3.9: icmp_seq=3 ttl=64 time=1.343 ms
64 bytes from 10.0.3.9: icmp_seq=4 ttl=64 time=1.434 ms
64 bytes from 10.0.3.9: icmp_seq=5 ttl=64 time=1.306 ms
^C
--- 10.0.3.9 ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.271/9.826/52.304/18.997 ms
```

```
user@R1> ping 10.0.3.10
```

```
PING 10.0.3.10 (10.0.3.10): 56 data bytes
64 bytes from 10.0.3.10: icmp_seq=0 ttl=64 time=1.431 ms
64 bytes from 10.0.3.10: icmp_seq=1 ttl=64 time=1.296 ms
64 bytes from 10.0.3.10: icmp_seq=2 ttl=64 time=1.887 ms
^C
--- 10.0.3.10 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.296/1.538/1.887/0.253 ms
```

```
user@R2> ping 10.0.2.9
```

```
PING 10.0.2.9 (10.0.2.9): 56 data bytes
```



```

64 bytes from 10.0.2.9: icmp_seq=0 ttl=64 time=1.365 ms
64 bytes from 10.0.2.9: icmp_seq=1 ttl=64 time=1.813 ms
64 bytes from 10.0.2.9: icmp_seq=2 ttl=64 time=1.290 ms
^C
--- 10.0.2.9 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.290/1.489/1.813/0.231 ms

```

```
user@R2> ping 10.0.2.10
```

```

PING 10.0.2.10 (10.0.2.10): 56 data bytes
64 bytes from 10.0.2.10: icmp_seq=0 ttl=63 time=1.318 ms
64 bytes from 10.0.2.10: icmp_seq=1 ttl=63 time=1.394 ms
64 bytes from 10.0.2.10: icmp_seq=2 ttl=63 time=1.366 ms
64 bytes from 10.0.2.10: icmp_seq=3 ttl=63 time=1.305 ms
^C
--- 10.0.2.10 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.305/1.346/1.394/0.036 ms

```

```
user@R3> ping 10.0.1.10
```

```

PING 10.0.1.10 (10.0.1.10): 56 data bytes
64 bytes from 10.0.1.10: icmp_seq=0 ttl=63 time=1.316 ms
64 bytes from 10.0.1.10: icmp_seq=1 ttl=63 time=1.418 ms
64 bytes from 10.0.1.10: icmp_seq=2 ttl=63 time=1.277 ms
^C
--- 10.0.1.10 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.277/1.337/1.418/0.059 ms

```

```
user@R3> ping 10.0.1.9
```

```

PING 10.0.1.9 (10.0.1.9): 56 data bytes
64 bytes from 10.0.1.9: icmp_seq=0 ttl=64 time=1.381 ms
64 bytes from 10.0.1.9: icmp_seq=1 ttl=64 time=1.499 ms
64 bytes from 10.0.1.9: icmp_seq=2 ttl=64 time=1.300 ms
64 bytes from 10.0.1.9: icmp_seq=3 ttl=64 time=1.397 ms
^C

```

```

--- 10.0.1.9 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.300/1.394/1.499/0.071 ms

```

## Meaning

Routers R1, R2, and R3 have a peer relationship with each other.

## Verifying That IS-IS Is Configured

### Purpose

Make sure that the IS-IS instance is running on Routers R1, R2, and R3, and that they are adjacent to each other.

### Action

Use the `show isis adjacency detail` command to check the adjacency between the routers.

Router R1

```
user@R1> show isis adjacency detail
```

R2

```

Interface: ge-0/0/0, Level: 1, State: Up, Expires in 8 secs
Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:23:59 ago
Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bd
Topologies: IPV4-Multicast
Restart capable: Yes, Adjacency advertisement: Advertise
LAN id: R2.02, IP addresses: 10.0.1.10

```

R2

```

Interface: ge-0/0/0, Level: 2, State: Up, Expires in 8 secs
Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:23:58 ago
Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bd
Topologies: IPV4-Multicast
Restart capable: Yes, Adjacency advertisement: Advertise
LAN id: R2.02, IP addresses: 10.0.1.10

```

R3

```
Interface: ge-1/0/0, Level: 1, State: Up, Expires in 7 secs
```

Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:24:20 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bd  
 Topologies: IPV4-Multicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R3.02, IP addresses: 10.0.2.10

### R3

Interface: ge-1/0/0, Level: 2, State: Up, Expires in 6 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:24:20 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bd  
 Topologies: IPV4-Multicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R3.02, IP addresses: 10.0.2.10

## Router R2

user@R2> **show isis adjacency detail**

### R1

Interface: ge-0/0/0, Level: 1, State: Up, Expires in 20 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:27:50 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bc  
 Topologies: IPV4-Multicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R2.02, IP addresses: 10.0.1.9

### R1

Interface: ge-0/0/0, Level: 2, State: Up, Expires in 26 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:27:50 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bc  
 Topologies: IPV4-Multicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R2.02, IP addresses: 10.0.1.9

### R3

Interface: ge-1/0/0, Level: 1, State: Up, Expires in 8 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:27:22 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bd  
 Topologies: IPV4-Multicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R3.03, IP addresses: 10.0.3.10

R3

Interface: ge-1/0/0, Level: 2, State: Up, Expires in 8 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:27:22 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bd  
 Topologies: IPV4-Multicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R3.03, IP addresses: 10.0.3.10

## Router R3

user@R3> **show isis adjacency detail**

R2

Interface: ge-0/0/0, Level: 1, State: Up, Expires in 18 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:33:09 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bc  
 Topologies: IPV4-Multicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R3.03, IP addresses: 10.0.3.9

R2

Interface: ge-0/0/0, Level: 2, State: Up, Expires in 22 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:33:09 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bc  
 Topologies: IPV4-Multicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R3.03, IP addresses: 10.0.3.9

R1

Interface: ge-1/0/0, Level: 1, State: Up, Expires in 21 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:33:59 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bc  
 Topologies: IPV4-Multicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R3.02, IP addresses: 10.0.2.9

R1

Interface: ge-1/0/0, Level: 2, State: Up, Expires in 19 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 2d 19:33:59 ago  
 Circuit type: 3, Speaks: IP, MAC address: 0:1b:c0:86:54:bc  
 Topologies: IPV4-Multicast

Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: R3.02, IP addresses: 10.0.2.9

## Meaning

IS-IS is configured on Routers R1, R2, and R3, and they are adjacent to each other.

## Verifying the Configured Multicast Metric Values

### Purpose

Make sure that the SPF calculations are accurate as per the configured multicast metric values on Routers R1, R2, and R3.

### Action

Use the `show isis spf results` command to check the SPF calculations for the network.

Router R1

```
user@R1> show isis spf results
...
IPV4 Multicast IS-IS level 1 SPF results:
Node  Metric  Interface  NH  Via  SNPA
R3.03  28         ge-1/0/0   IPV4 R3  0:1b:c0:86:54:bd
R2.00  18         ge-0/0/0   IPV4 R2  0:1b:c0:86:54:bd
R3.00  17         ge-1/0/0   IPV4 R3  0:1b:c0:86:54:bd
R1.00  0
4 nodes

IPV4 Multicast IS-IS level 2 SPF results:
Node  Metric  Interface  NH  Via  SNPA
R3.03  40         ge-0/0/0   IPV4 R2  0:1b:c0:86:54:bd
R3.00  22         ge-1/0/0   IPV4 R3  0:1b:c0:86:54:bd
R2.00  14         ge-0/0/0   IPV4 R2  0:1b:c0:86:54:bd
R1.00  0
4 nodes
```

## Router R2

```

user@R2> show isis spf results
...
IPV4 Multicast IS-IS level 1 SPF results:
Node  Metric    Interface    NH  Via  SNPA
R3.02  29           ge-0/0/0     IPV4 R1  0:1b:c0:86:54:bc
R3.00  18           ge-1/0/0     IPV4 R3  0:1b:c0:86:54:bd
R1.00  12           ge-0/0/0     IPV4 R1  0:1b:c0:86:54:bc
R2.02  12
R2.00  0
      5 nodes

IPV4 Multicast IS-IS level 2 SPF results:
Node  Metric    Interface    NH  Via  SNPA
R3.02  45           ge-0/0/0     IPV4 R1  0:1b:c0:86:54:bc
R3.00  26           ge-1/0/0     IPV4 R3  0:1b:c0:86:54:bd
R1.00  23           ge-0/0/0     IPV4 R1  0:1b:c0:86:54:bc
R2.02  23
R2.00  0
      5 nodes

```

## Router R3

```

user@R3> show isis spf results
...
IPV4 Multicast IS-IS level 1 SPF results:
Node  Metric    Interface    NH  Via  SNPA
R3.02  26
R1.00  23           ge-0/0/0     IPV4 R2  0:1b:c0:86:54:bc
R2.02  23           ge-0/0/0     IPV4 R2  0:1b:c0:86:54:bc
R2.00  11           ge-0/0/0     IPV4 R2  0:1b:c0:86:54:bc
R3.03  11
R3.00  0
      6 nodes

IPV4 Multicast IS-IS level 2 SPF results:
Node  Metric    Interface    NH  Via  SNPA
R2.02  34           ge-1/0/0     IPV4 R1  0:1b:c0:86:54:bc
R2.00  21           ge-0/0/0     IPV4 R2  0:1b:c0:86:54:bc
R3.03  21

```

```

R1.00 20          ge-1/0/0      IPV4 R1    0:1b:c0:86:54:bc
R3.02 20
R3.00 0
      6 nodes

```

## Meaning

The configured multicast metric values are used in SPF calculations for the IS-IS network.

## Verifying the Configuration of the Multicast Topology

### Purpose

Make sure that the multicast topology is configured on Routers R1, R2, and R3.

### Action

Use the `show isis database detail` command to verify the multicast topology configuration on the routers.

Router R1

```

user@R1> show isis database detail

IS-IS level 1 link-state database:

R1.00-00 Sequence: 0x142, Checksum: 0xd07, Lifetime: 663 secs
  IPV4 Unicast IS neighbor: R2.02      Metric:      15
  IPV4 Unicast IS neighbor: R3.02      Metric:      15
  IPV4 Multicast IS neighbor: R2.02     Metric:      18
  IPV4 Multicast IS neighbor: R3.02     Metric:      17
  IP IPV4 Unicast prefix: 10.0.1.8/30  Metric:      15 Internal Up
  IP IPV4 Unicast prefix: 10.0.2.8/30  Metric:      15 Internal Up

R2.00-00 Sequence: 0x13f, Checksum: 0xf02b, Lifetime: 883 secs
  IPV4 Unicast IS neighbor: R2.02      Metric:      13
  IPV4 Unicast IS neighbor: R3.03      Metric:      14
  IPV4 Multicast IS neighbor: R2.02     Metric:      12
  IPV4 Multicast IS neighbor: R3.03     Metric:      18
  IP IPV4 Unicast prefix: 10.0.1.8/30  Metric:      13 Internal Up
  IP IPV4 Unicast prefix: 10.0.3.8/30  Metric:      14 Internal Up

```

R2.02-00 Sequence: 0x13c, Checksum: 0x57e2, Lifetime: 913 secs

IPV4 Unicast IS neighbor: R1.00 Metric: 0

IPV4 Unicast IS neighbor: R2.00 Metric: 0

R3.00-00 Sequence: 0x13c, Checksum: 0xc8de, Lifetime: 488 secs

IPV4 Unicast IS neighbor: R3.02 Metric: 16

IPV4 Unicast IS neighbor: R3.03 Metric: 19

IPV4 Multicast IS neighbor: R3.02 Metric: 26

IPV4 Multicast IS neighbor: R3.03 Metric: 11

IP IPV4 Unicast prefix: 10.0.2.8/30 Metric: 16 Internal Up

IP IPV4 Unicast prefix: 10.0.3.8/30 Metric: 19 Internal Up

R3.02-00 Sequence: 0x139, Checksum: 0xfb0e, Lifetime: 625 secs

IPV4 Unicast IS neighbor: R1.00 Metric: 0

IPV4 Unicast IS neighbor: R3.00 Metric: 0

R3.03-00 Sequence: 0x138, Checksum: 0xad56, Lifetime: 714 secs

IPV4 Unicast IS neighbor: R2.00 Metric: 0

IPV4 Unicast IS neighbor: R3.00 Metric: 0

IS-IS level 2 link-state database:

R1.00-00 Sequence: 0x142, Checksum: 0x2c7c, Lifetime: 816 secs

IPV4 Unicast IS neighbor: R2.02 Metric: 20

IPV4 Unicast IS neighbor: R3.02 Metric: 31

IPV4 Multicast IS neighbor: R2.02 Metric: 14

IPV4 Multicast IS neighbor: R3.02 Metric: 22

IP IPV4 Unicast prefix: 10.0.1.8/30 Metric: 20 Internal Up

IP IPV4 Unicast prefix: 10.0.2.8/30 Metric: 31 Internal Up

IP IPV4 Unicast prefix: 10.0.3.8/30 Metric: 29 Internal Up

R2.00-00 Sequence: 0x13f, Checksum: 0x4826, Lifetime: 966 secs

IPV4 Unicast IS neighbor: R2.02 Metric: 29

IPV4 Unicast IS neighbor: R3.03 Metric: 32

IPV4 Multicast IS neighbor: R2.02 Metric: 23

IPV4 Multicast IS neighbor: R3.03 Metric: 26

IP IPV4 Unicast prefix: 10.0.1.8/30 Metric: 29 Internal Up

IP IPV4 Unicast prefix: 10.0.2.8/30 Metric: 28 Internal Up

IP IPV4 Unicast prefix: 10.0.3.8/30 Metric: 32 Internal Up

R2.02-00 Sequence: 0x13c, Checksum: 0x57e2, Lifetime: 966 secs

IPV4 Unicast IS neighbor: R1.00 Metric: 0

IPV4 Unicast IS neighbor: R2.00 Metric: 0



```

R3.00-00 Sequence: 0x13d, Checksum: 0x1b19, Lifetime: 805 secs
  IPV4 Unicast IS neighbor: R3.02      Metric:      30
  IPV4 Unicast IS neighbor: R3.03      Metric:      27
  IPV4 Multicast IS neighbor: R3.02     Metric:      20
  IPV4 Multicast IS neighbor: R3.03     Metric:      21
  IP IPV4 Unicast prefix: 10.0.1.8/30  Metric:      31 Internal Up
  IP IPV4 Unicast prefix: 10.0.2.8/30  Metric:      30 Internal Up
  IP IPV4 Unicast prefix: 10.0.3.8/30  Metric:      27 Internal Up

R3.02-00 Sequence: 0x139, Checksum: 0xfb0e, Lifetime: 844 secs
  IPV4 Unicast IS neighbor: R1.00      Metric:       0
  IPV4 Unicast IS neighbor: R3.00      Metric:       0

R3.03-00 Sequence: 0x139, Checksum: 0xab57, Lifetime: 844 secs
  IPV4 Unicast IS neighbor: R2.00      Metric:       0
  IPV4 Unicast IS neighbor: R3.00      Metric:       0

```

## Router R2

```

user@R2> show isis database detail

IS-IS level 1 link-state database:

R1.00-00 Sequence: 0x142, Checksum: 0xd07, Lifetime: 524 secs
  IPV4 Unicast IS neighbor: R2.02      Metric:      15
  IPV4 Unicast IS neighbor: R3.02      Metric:      15
  IPV4 Multicast IS neighbor: R2.02     Metric:      18
  IPV4 Multicast IS neighbor: R3.02     Metric:      17
  IP IPV4 Unicast prefix: 10.0.1.8/30  Metric:      15 Internal Up
  IP IPV4 Unicast prefix: 10.0.2.8/30  Metric:      15 Internal Up

R2.00-00 Sequence: 0x13f, Checksum: 0xf02b, Lifetime: 748 secs
  IPV4 Unicast IS neighbor: R2.02      Metric:      13
  IPV4 Unicast IS neighbor: R3.03      Metric:      14
  IPV4 Multicast IS neighbor: R2.02     Metric:      12
  IPV4 Multicast IS neighbor: R3.03     Metric:      18
  IP IPV4 Unicast prefix: 10.0.1.8/30  Metric:      13 Internal Up
  IP IPV4 Unicast prefix: 10.0.3.8/30  Metric:      14 Internal Up

R2.02-00 Sequence: 0x13c, Checksum: 0x57e2, Lifetime: 777 secs
  IPV4 Unicast IS neighbor: R1.00      Metric:       0

```

IPv4 Unicast IS neighbor: R2.00      Metric:      0

R3.00-00 Sequence: 0x13d, Checksum: 0xc6df, Lifetime: 1102 secs

IPv4 Unicast IS neighbor: R3.02      Metric:      16

IPv4 Unicast IS neighbor: R3.03      Metric:      19

IPv4 Multicast IS neighbor: R3.02      Metric:      26

IPv4 Multicast IS neighbor: R3.03      Metric:      11

IP IPv4 Unicast prefix: 10.0.2.8/30      Metric:      16 Internal Up

IP IPv4 Unicast prefix: 10.0.3.8/30      Metric:      19 Internal Up

R3.02-00 Sequence: 0x139, Checksum: 0xfb0e, Lifetime: 488 secs

IPv4 Unicast IS neighbor: R1.00      Metric:      0

IPv4 Unicast IS neighbor: R3.00      Metric:      0

R3.03-00 Sequence: 0x138, Checksum: 0xad56, Lifetime: 577 secs

IPv4 Unicast IS neighbor: R2.00      Metric:      0

IPv4 Unicast IS neighbor: R3.00      Metric:      0

IS-IS level 2 link-state database:

R1.00-00 Sequence: 0x142, Checksum: 0x2c7c, Lifetime: 676 secs

IPv4 Unicast IS neighbor: R2.02      Metric:      20

IPv4 Unicast IS neighbor: R3.02      Metric:      31

IPv4 Multicast IS neighbor: R2.02      Metric:      14

IPv4 Multicast IS neighbor: R3.02      Metric:      22

IP IPv4 Unicast prefix: 10.0.1.8/30      Metric:      20 Internal Up

IP IPv4 Unicast prefix: 10.0.2.8/30      Metric:      31 Internal Up

IP IPv4 Unicast prefix: 10.0.3.8/30      Metric:      29 Internal Up

R2.00-00 Sequence: 0x13f, Checksum: 0x4826, Lifetime: 831 secs

IPv4 Unicast IS neighbor: R2.02      Metric:      29

IPv4 Unicast IS neighbor: R3.03      Metric:      32

IPv4 Multicast IS neighbor: R2.02      Metric:      23

IPv4 Multicast IS neighbor: R3.03      Metric:      26

IP IPv4 Unicast prefix: 10.0.1.8/30      Metric:      29 Internal Up

IP IPv4 Unicast prefix: 10.0.2.8/30      Metric:      28 Internal Up

IP IPv4 Unicast prefix: 10.0.3.8/30      Metric:      32 Internal Up

R2.02-00 Sequence: 0x13c, Checksum: 0x57e2, Lifetime: 831 secs

IPv4 Unicast IS neighbor: R1.00      Metric:      0

IPv4 Unicast IS neighbor: R2.00      Metric:      0

R3.00-00 Sequence: 0x13d, Checksum: 0x1b19, Lifetime: 667 secs

```

IPV4 Unicast IS neighbor: R3.02      Metric:      30
IPV4 Unicast IS neighbor: R3.03      Metric:      27
IPV4 Multicast IS neighbor: R3.02     Metric:      20
IPV4 Multicast IS neighbor: R3.03     Metric:      21
IP IPV4 Unicast prefix: 10.0.1.8/30  Metric:      31 Internal Up
IP IPV4 Unicast prefix: 10.0.2.8/30  Metric:      30 Internal Up
IP IPV4 Unicast prefix: 10.0.3.8/30  Metric:      27 Internal Up

R3.02-00 Sequence: 0x139, Checksum: 0xfb0e, Lifetime: 707 secs
  IPV4 Unicast IS neighbor: R1.00     Metric:      0
  IPV4 Unicast IS neighbor: R3.00     Metric:      0

R3.03-00 Sequence: 0x139, Checksum: 0xab57, Lifetime: 707 secs
  IPV4 Unicast IS neighbor: R2.00     Metric:      0
  IPV4 Unicast IS neighbor: R3.00     Metric:      0

```

### Router R3

```

user@R3> show isis database detail

IS-IS level 1 link-state database:

R1.00-00 Sequence: 0x143, Checksum: 0xb08, Lifetime: 1155 secs
  IPV4 Unicast IS neighbor: R2.02     Metric:      15
  IPV4 Unicast IS neighbor: R3.02     Metric:      15
  IPV4 Multicast IS neighbor: R2.02    Metric:      18
  IPV4 Multicast IS neighbor: R3.02    Metric:      17
  IP IPV4 Unicast prefix: 10.0.1.8/30  Metric:      15 Internal Up
  IP IPV4 Unicast prefix: 10.0.2.8/30  Metric:      15 Internal Up

R2.00-00 Sequence: 0x13f, Checksum: 0xf02b, Lifetime: 687 secs
  IPV4 Unicast IS neighbor: R2.02     Metric:      13
  IPV4 Unicast IS neighbor: R3.03     Metric:      14
  IPV4 Multicast IS neighbor: R2.02    Metric:      12
  IPV4 Multicast IS neighbor: R3.03    Metric:      18
  IP IPV4 Unicast prefix: 10.0.1.8/30  Metric:      13 Internal Up
  IP IPV4 Unicast prefix: 10.0.3.8/30  Metric:      14 Internal Up

R2.02-00 Sequence: 0x13c, Checksum: 0x57e2, Lifetime: 716 secs
  IPV4 Unicast IS neighbor: R1.00     Metric:      0
  IPV4 Unicast IS neighbor: R2.00     Metric:      0

```

R3.00-00 Sequence: 0x13d, Checksum: 0xc6df, Lifetime: 1044 secs

IPv4 Unicast IS neighbor: R3.02	Metric:	16
IPv4 Unicast IS neighbor: R3.03	Metric:	19
IPv4 Multicast IS neighbor: R3.02	Metric:	26
IPv4 Multicast IS neighbor: R3.03	Metric:	11
IP IPv4 Unicast prefix: 10.0.2.8/30	Metric:	16 Internal Up
IP IPv4 Unicast prefix: 10.0.3.8/30	Metric:	19 Internal Up

R3.02-00 Sequence: 0x139, Checksum: 0xfb0e, Lifetime: 430 secs

IPv4 Unicast IS neighbor: R1.00	Metric:	0
IPv4 Unicast IS neighbor: R3.00	Metric:	0

R3.03-00 Sequence: 0x138, Checksum: 0xad56, Lifetime: 519 secs

IPv4 Unicast IS neighbor: R2.00	Metric:	0
IPv4 Unicast IS neighbor: R3.00	Metric:	0

IS-IS level 2 link-state database:

R1.00-00 Sequence: 0x142, Checksum: 0x2c7c, Lifetime: 617 secs

IPv4 Unicast IS neighbor: R2.02	Metric:	20
IPv4 Unicast IS neighbor: R3.02	Metric:	31
IPv4 Multicast IS neighbor: R2.02	Metric:	14
IPv4 Multicast IS neighbor: R3.02	Metric:	22
IP IPv4 Unicast prefix: 10.0.1.8/30	Metric:	20 Internal Up
IP IPv4 Unicast prefix: 10.0.2.8/30	Metric:	31 Internal Up
IP IPv4 Unicast prefix: 10.0.3.8/30	Metric:	29 Internal Up

R2.00-00 Sequence: 0x13f, Checksum: 0x4826, Lifetime: 769 secs

IPv4 Unicast IS neighbor: R2.02	Metric:	29
IPv4 Unicast IS neighbor: R3.03	Metric:	32
IPv4 Multicast IS neighbor: R2.02	Metric:	23
IPv4 Multicast IS neighbor: R3.03	Metric:	26
IP IPv4 Unicast prefix: 10.0.1.8/30	Metric:	29 Internal Up
IP IPv4 Unicast prefix: 10.0.2.8/30	Metric:	28 Internal Up
IP IPv4 Unicast prefix: 10.0.3.8/30	Metric:	32 Internal Up

R2.02-00 Sequence: 0x13c, Checksum: 0x57e2, Lifetime: 769 secs

IPv4 Unicast IS neighbor: R1.00	Metric:	0
IPv4 Unicast IS neighbor: R2.00	Metric:	0

R3.00-00 Sequence: 0x13d, Checksum: 0x1b19, Lifetime: 610 secs

IPv4 Unicast IS neighbor: R3.02	Metric:	30
IPv4 Unicast IS neighbor: R3.03	Metric:	27

```

IPv4 Multicast IS neighbor: R3.02      Metric:      20
IPv4 Multicast IS neighbor: R3.03      Metric:      21
IP IPv4 Unicast prefix: 10.0.1.8/30    Metric:      31 Internal Up
IP IPv4 Unicast prefix: 10.0.2.8/30    Metric:      30 Internal Up
IP IPv4 Unicast prefix: 10.0.3.8/30    Metric:      27 Internal Up

R3.02-00 Sequence: 0x139, Checksum: 0xfb0e, Lifetime: 649 secs
  IPv4 Unicast IS neighbor: R1.00      Metric:      0
  IPv4 Unicast IS neighbor: R3.00      Metric:      0

R3.03-00 Sequence: 0x139, Checksum: 0xab57, Lifetime: 649 secs
  IPv4 Unicast IS neighbor: R2.00      Metric:      0
  IPv4 Unicast IS neighbor: R3.00      Metric:      0

```

## Meaning

Multicast topology is configured on Routers R1, R2, and R3.

## RELATED DOCUMENTATION

[IS-IS Multicast Topologies Overview](#) | 287

## Understanding Dual Stacking of IPv4 and IPv6 Unicast Addresses

Service providers and enterprises are faced with growing their networks using IPv6, while continuing to serve IPv4 customers.

Increasingly, the public side of network address translation (NAT) devices is IPv6 rather than IPv4. Service providers cannot continue giving customers globally routable IPv4 addresses, they cannot get new globally routable IPv4 addresses for expanding their own networks, and yet they must continue to serve both IPv4 customers and new customers, all of whom are primarily trying to reach IPv4 destinations.

IPv4 and IPv6 must coexist for some number of years, and their coexistence must be transparent to end users. If an IPv4-to-IPv6 transition is successful, end users should not even notice it.

A dual-stack device is a device with network interfaces that can originate and understand both IPv4 and IPv6 packets.

Other strategies, such as manually or dynamically configured tunnels and translation devices exist, but dual stacking is often the preferable solution in many scenarios. The dual-stacked device can interoperate equally with IPv4 devices, IPv6 devices, and other dual-stacked devices. When both devices are dual stacked, the two devices agree on which IP version to use.

The transition is driven by DNS. If a dual-stacked device queries the name of a destination and DNS gives it an IPv4 address (a DNS A Record), it sends IPv4 packets. If DNS responds with an IPv6 address (a DNS AAAA Record), it sends IPv6 packets.

Keep in mind that if you are going to dual stack all of your network devices, the interfaces need both an IPv6 and an IPv4 address. This raises the issue that the Internet has run out of IPv4 addresses, which is the main reason IPv6 is needed in the first place. If you do not have an abundant supply of IPv4 addresses to apply to your devices, you can still use dual stacking, but you will need to conserve your supply of IPv4 addresses by using network address translation (NAT). Building dual-stacked networks with a mix of global IPv6 addresses and NAT-ed IPv4 addresses is quite feasible. Some specific solutions include carrier-grade NAT (CGN), NAT44(4), NAT64, NAT464, and dual-stack lite.

[Table 5 on page 312](#) describes at a high level how to pick a network addressing technique. In reality, a complete solution might include a set of techniques to satisfy multiple service needs. It is important to understand the backbone technology being used on the network and also to know if the provider has control over the access customer premises equipment (CPE).

**Table 5: Choosing the Right Solution to Address Next-Generation Addressing Requirements**

CPE Network	Access Network	Destinations	Solution
IPv4	IPv4	IPv4 Internet	NAT44(4)
IPv4/ IPv6	IPv6	IPv4 Internet	DS-Lite with NAT44
IPv4/ IPv6	IPv4	IPv6 Internet	6rd (6to4)
IPv4	IPv6	IPv4 Internet	NAT64

## RELATED DOCUMENTATION

*Understanding IPv6 Dual-Stack Lite*

## Example: Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses

### IN THIS SECTION

- [Requirements | 313](#)
- [Overview | 313](#)
- [Configuration | 314](#)
- [Verification | 318](#)

This example shows how to configure IPv4 and IPv6 dual stacking in IS-IS.

### Requirements

No special configuration beyond device initialization is required before configuring this example.

### Overview



Video: [IS-IS Dual Stacking](#)

You can use IPv4 and IPv6 dual stacking to begin your migration from IPv4 to IPv6 by implementing IPv6 alongside IPv4 in your existing networks. This allows you to implement IPv6 so that you can provide the same services over IPv6—for example, video, voice, high-quality data—that you currently provide in your IPv4 networks. You can then perform incremental upgrades to IPv6 and avoid service disruptions while migrating from IPv4 to IPv6.

Unlike RIP and OSPF, IS-IS does not require a distinct protocol or a new version to support IPv6. Because IS-IS uses ISO addresses, the configuration for IPv6 and IPv4 is identical in the Junos OS implementation of IS-IS. For IS-IS to carry IPv6 routes, you only need to add IPv6 addresses to IS-IS enabled interfaces or include other IPv6 routes in your IS-IS export policy.

The only explicit configuration needed in IS-IS with regard to IPv6 is if you want to disable it. Alternatively, you can disable IPv4 routing and use IS-IS with IPv6 only. An example of each is provided here:

Disable IPv6 routing in IS-IS:

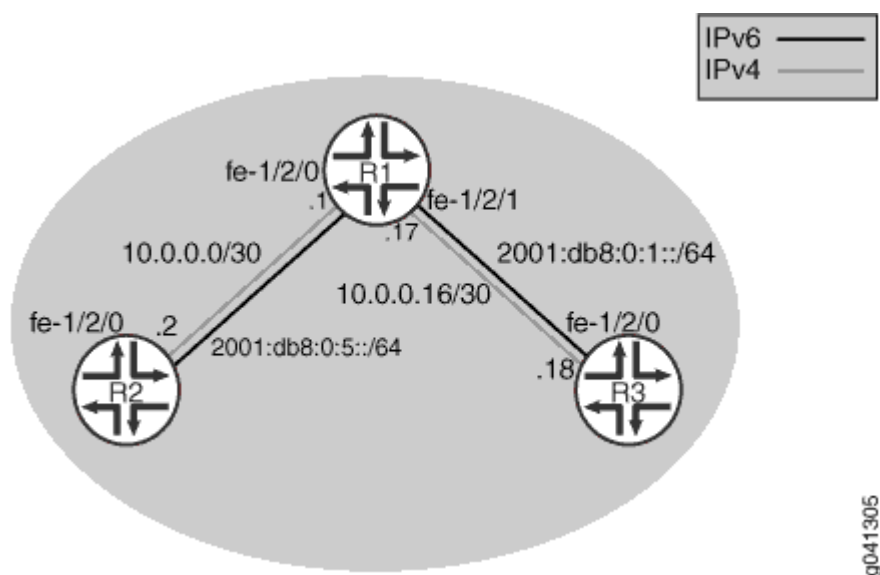
```
[edit protocols isis]
user@host# set no-ipv6-routing
```

Use IS-IS exclusively for IPv6 routing:

```
[edit protocols isis]
user@host# set no-ipv4-routing
```

Figure 25 on page 314 shows the topology used in this example.

Figure 25: IS-IS IPv4 and IPv6 Dual Stacking Topology



"CLI Quick Configuration" on page 315 shows the configuration for all of the devices in Figure 25 on page 314. The section "No Link Title" on page 316 describes the steps on Device R1.

## Configuration

### IN THIS SECTION

- Procedure | 315



## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R1

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.1/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.17/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::1/128
set protocols isis interface fe-1/2/0.0
set protocols isis interface fe-1/2/1.0
set protocols isis interface lo0.0
```

#### Device R2

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.2/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0002.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::2/128
set protocols isis interface fe-1/2/0.0
set protocols isis interface lo0.0
```

#### Device R3

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.18/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0003.00
```

```

set interfaces lo0 unit 0 family inet6 address 2001:db8::3/128
set protocols isis interface fe-1/2/0.0
set protocols isis interface lo0.0

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the Junos OS [CLI User Guide](#).

To configure IS-IS dual stacking:

1. Configure the interfaces, including both IPv4 and IPv6 addresses on each interface.

Optionally, include the `eui-64` statement to automatically generate the host number portion of interface addresses.

```

[edit interfaces]
user@R1# set fe-1/2/0 unit 0 family inet address 10.0.0.1/30
user@R1# set fe-1/2/0 unit 0 family iso
user@R1# set fe-1/2/0 unit 0 family inet6 address 2001:db8:0:5::/64 eui-64
user@R1# set fe-1/2/1 unit 0 family inet address 10.0.0.17/30
user@R1# set fe-1/2/1 unit 0 family iso
user@R1# set fe-1/2/1 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
user@R1# set lo0 unit 0 family inet address 192.168.0.1/32
user@R1# set lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
user@R1# set lo0 unit 0 family inet6 address 2001:db8::1/128

```

2. Enable IS-IS on the interfaces.

```

[edit protocols isis]
user@R1# set interface fe-1/2/0.0
user@R1# set interface fe-1/2/1.0
user@R1# set interface lo0.0

```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces` and `show protocols` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces
fe-1/2/0 {
  unit 0 {
    family inet {
      address 10.0.0.1/30;
    }
    family iso;
    family inet6 {
      address 2001:db8:0:5::/64 {
        eui-64;
      }
    }
  }
}
fe-1/2/1 {
  unit 0 {
    family inet {
      address 10.0.0.17/30;
    }
    family iso;
    family inet6 {
      address 2001:db8:0:1::/64 {
        eui-64;
      }
    }
  }
}
lo0 {
  unit 0 {
    family inet {
      address 192.168.0.1/32;
    }
    family iso {
      address 49.0002.0192.0168.0001.00;
    }
    family inet6 {
```

```

        address 2001:db8::1/128;
    }
}

```

```

user@R1# show protocols
isis {
    interface fe-1/2/0.0;
    interface fe-1/2/1.0;
    interface lo0.0;
}

```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Checking the Neighbor Adjacencies | 318](#)
- [Pinging the IPv6 Interfaces | 319](#)
- [Checking the IPv6 Routing Table | 320](#)

Confirm that the configuration is working properly.

### Checking the Neighbor Adjacencies

#### Purpose

Determine what topologies are supported on neighboring IS-IS devices.

#### Action

From operational mode, enter the `show isis adjacency detail` command.

```

user@R1> show isis adjacency detail

```

```

R2

```

```

Interface: fe-1/2/0.0, Level: 3, State: Up, Expires in 24 secs
Priority: 0, Up/Down transitions: 1, Last transition: 18:34:08 ago
Circuit type: 3, Speaks: IP, IPv6
Topologies: Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 10.0.0.2
IPv6 addresses: fe80::2a0:a514:0:24c

```

R3

```

Interface: fe-1/2/1.0, Level: 3, State: Up, Expires in 21 secs
Priority: 0, Up/Down transitions: 1, Last transition: 18:33:41 ago
Circuit type: 3, Speaks: IP, IPv6
Topologies: Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 10.0.0.18
IPv6 addresses: fe80::2a0:a514:0:124c

```

## Meaning

As expected, the output shows that the two neighbors support both IPv4 and IPv6. The IPv4 address and the IPv6 link-local address are also shown.

## Pinging the IPv6 Interfaces

## Purpose

Make sure that you can ping the remote IPv6 interfaces.

## Action

From operational mode, enter the ping command to ping from Device R2 to Device R3.

1. Determine the IPv6 address assigned to Device R3.

If you use EUI-64 addressing as shown in the example, the host portion of the IPv6 addresses is assigned automatically. To determine what addresses are assigned, use the `show interfaces terse` command on Device R3.

```

user@R3> show interfaces terse
Interface          Admin Link Proto  Local          Remote
fe-1/2/0

```

```

fe-1/2/0.0          up    up    inet    10.0.0.18/30
                    iso
                    inet6    2001:db8:0:1:2a0:a514:0:124c/64
                    fe80::2a0:a514:0:124c/64

lo0
lo0.0               up    up    inet    192.168.0.3      --> 0/0
                    iso    49.0002.0192.0168.0003
                    inet6    2001:db8::3
                    fe80::2a0:a50f:fc56:14c

```

The IPv6 addresses that should be pingable are 2001:db8:0:1:2a0:a514:0:124c and 2001:db8::3.

2. From Device R2, ping the Device R3 fe-1/2/0.0 IPv6 interface address and the lo0.0 IPv6 interface address.

```

user@R2> ping 2001:db8:0:1:2a0:a514:0:124c
PING6(56=40+8+8 bytes) 2001:db8:0:5:2a0:a514:0:24c --> 2001:db8:0:1:2a0:a514:0:124c
16 bytes from 2001:db8:0:1:2a0:a514:0:124c, icmp_seq=0 hlim=63 time=2.373 ms
16 bytes from 2001:db8:0:1:2a0:a514:0:124c, icmp_seq=1 hlim=63 time=1.600 ms
16 bytes from 2001:db8:0:1:2a0:a514:0:124c, icmp_seq=2 hlim=63 time=2.228 ms

```

```

user@R2> ping 2001:db8::3
PING6(56=40+8+8 bytes) 2001:db8:0:5:2a0:a514:0:24c --> 2001:db8::3
16 bytes from 2001:db8::3, icmp_seq=0 hlim=63 time=1.797 ms
16 bytes from 2001:db8::3, icmp_seq=1 hlim=63 time=1.430 ms
16 bytes from 2001:db8::3, icmp_seq=2 hlim=63 time=2.525 ms

```

## Meaning

This test confirms that IS-IS has learned the IPv6 routes.

## Checking the IPv6 Routing Table

## Purpose

Verify that the expected routes are in the IPv6 routing table.

## Action

```

user@R1> show route table inet6.0
inet6.0: 11 destinations, 12 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::1/128    *[Direct/0] 18:52:52
                  >   via lo0.0
2001:db8::2/128    *[IS-IS/15] 01:59:52, metric 10
                  >   to fe80::2a0:a514:0:24c via fe-1/2/0.0
2001:db8::3/128    *[IS-IS/15] 01:59:52, metric 10
                  >   to fe80::2a0:a514:0:124c via fe-1/2/1.0
2001:db8:0:1::/64  *[Direct/0] 18:52:15
                  >   via fe-1/2/1.0
2001:db8:0:1:2a0:a514:0:114c/128
                  *[Local/0] 18:52:48
                  Local via fe-1/2/1.0
2001:db8:0:5::/64  *[Direct/0] 18:52:49
                  >   via fe-1/2/0.0
2001:db8:0:5:2a0:a514:0:14c/128
                  *[Local/0] 18:52:49
                  Local via fe-1/2/0.0
fe80::/64          *[Direct/0] 18:52:49
                  >   via fe-1/2/0.0
                  [Direct/0] 18:52:15
                  >   via fe-1/2/1.0
fe80::2a0:a50f:fc56:14c/128
                  *[Direct/0] 18:52:52
                  >   via lo0.0
fe80::2a0:a514:0:14c/128
                  *[Local/0] 18:52:49
                  Local via fe-1/2/0.0
fe80::2a0:a514:0:114c/128
                  *[Local/0] 18:52:48
                  Local via fe-1/2/1.0

```

## Meaning

The output shows the IPv6 interface routes (direct and local) and the IPv6 routes learned through IS-IS.

## RELATED DOCUMENTATION

[Example: Configuring IS-IS IPv4 and IPv6 Unicast Topologies](#) | 323

## Understanding IS-IS IPv4 and IPv6 Unicast Topologies

You can configure IS-IS to calculate an alternate IPv6 unicast topology, in addition to the normal IPv4 unicast topology, and add the corresponding routes to inet6.0. The IS-IS interface metrics for the IPv4 topology can be configured independently of the IPv6 metrics. You can also selectively disable interfaces from participating in the IPv6 topology while continuing to participate in the IPv4 topology. This enables you to exercise control over the paths that unicast data takes through a network.

A topology is the set of joined nodes. IS-IS evaluates all the paths in a single topology for each IS-IS level and uses the shortest-path-first (SPF) algorithm to determine the best path among all the feasible paths. Topology discovery and SPF calculation is performed in a protocol-neutral fashion because it is done at Layer 2 of the OSI model. If you load the topology with reachability information for a certain protocol (for example, IP), the assumption is that the circuits that are supposed to provide reachability between routing devices can carry the protocol. The SPF algorithm has a per-link orientation, not a per-address family or per-protocol orientation.

Multitopology routing enables you to override this default behavior by enabling a per-address family, per-protocol SPF calculation.

The additional CPU load associated with multiple runs of the SPF algorithm is generally not an issue with the processing power available on today's routing device control planes.

The multitopology extensions alter existing type, length, and value (TLV) tuples by adding a topology ID. Each routing device in a given topology maintains its adjacencies and runs a per-topology SPF calculation.

## RELATED DOCUMENTATION

[Example: Configuring IS-IS IPv4 and IPv6 Unicast Topologies](#) | 323



## Example: Configuring IS-IS IPv4 and IPv6 Unicast Topologies

### IN THIS SECTION

- [Requirements | 323](#)
- [Overview | 323](#)
- [Configuration | 325](#)
- [Verification | 331](#)

This example shows how to configure IS-IS to calculate an alternate IPv6 unicast topology, in addition to the normal IPv4 unicast topology.

### Requirements

No special configuration beyond device initialization is required before configuring this example.

### Overview

This example focuses on IPv4 and IPv6 unicast topologies. The IS-IS interface metrics for the IPv4 topology can be configured independently of the IPv6 metrics. You can also selectively disable interfaces from participating in the IPv6 topology while continuing to participate in the IPv4 topology. This enables you to exercise control over the paths that unicast data takes through a network.

To enable an IPv6 unicast topology for IS-IS, include the `ipv6-unicast` statement:

```
isis {  
    topologies {  
        ipv6-unicast;  
    }  
}
```

To configure a metric for the IPv6 unicast topology, include the `ipv6-unicast-metric` statement:

```
isis {  
    interface interface-name {  
        level level-number {  
            ipv6-unicast-metric number;  
        }  
    }  
}
```

```
    }
}
```

To exclude an interface from the IPv6 unicast topologies for IS-IS, include the `no-ipv6-unicast` statement:

```
isis {
    interface interface-name {
        no-ipv6-unicast;
    }
}
```

[Figure 26 on page 325](#) shows the topology used in this example. The black lines indicate link membership in the IPv6 topology. The gray lines indicate membership to the IPv4 topology. Using regular TLVs, it would not be possible to build multiple topologies and run an SPF calculation based on them. The multitopology extensions describe an extension to carry the set of supported protocols in the hello packet. After activating multitopology routing support on a link, the link carries all the topologies that the underlying circuit is able to relay.



## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R1

```
set interfaces so-1/2/0 unit 0 family inet address 10.0.0.1/30
set interfaces so-1/2/0 unit 0 family iso
set interfaces so-1/2/0 unit 0 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces so-1/2/1 unit 0 family inet address 10.0.0.13/30
set interfaces so-1/2/1 unit 0 family iso
set interfaces so-1/2/2 unit 0 family inet address 10.0.0.17/30
set interfaces so-1/2/2 unit 0 family iso
set interfaces so-1/2/2 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::1/128
set protocols isis topologies ipv6-unicast
set protocols isis interface so-1/2/0.0
set protocols isis interface so-1/2/1.0 no-ipv6-unicast
set protocols isis interface so-1/2/2.0
set protocols isis interface lo0.0
```

#### Device R2

```
set interfaces so-1/2/0 unit 0 family inet address 10.0.0.2/30
set interfaces so-1/2/0 unit 0 family iso
set interfaces so-1/2/0 unit 0 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces so-1/2/1 unit 0 family inet address 10.0.0.9/30
set interfaces so-1/2/1 unit 0 family iso
set interfaces so-1/2/1 unit 0 family inet6 address 2001:db8:0:4::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0002.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::2/128
set protocols isis topologies ipv6-unicast
set protocols isis interface so-1/2/0.2
```

```
set protocols isis interface so-1/2/1.0
set protocols isis interface lo0.0
```

### Device R3

```
set interfaces so-1/2/0 unit 0 family inet address 10.0.0.18/30
set interfaces so-1/2/0 unit 0 family iso
set interfaces so-1/2/0 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces so-1/2/1 unit 0 family inet address 10.0.0.25/30
set interfaces so-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0003.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::3/128
set protocols isis topologies ipv6-unicast
set protocols isis interface so-1/2/0.0
set protocols isis interface so-1/2/1.0 no-ipv6-unicast
set protocols isis interface lo0.0
```

### Device R4

```
set interfaces so-1/2/0 unit 0 family inet address 10.0.0.5/30
set interfaces so-1/2/0 unit 0 family iso
set interfaces so-1/2/0 unit 0 family inet6 address 2001:db8:0:2::/64 eui-64
set interfaces so-1/2/1 unit 0 family inet address 10.0.0.10/30
set interfaces so-1/2/1 unit 0 family iso
set interfaces so-1/2/1 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces so-1/2/2 unit 0 family inet address 10.0.0.21/30
set interfaces so-1/2/2 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.4/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0004.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::4/128
set protocols isis topologies ipv6-unicast
set protocols isis interface so-1/2/0.0
set protocols isis interface so-1/2/1.0
set protocols isis interface so-1/2/2.0 no-ipv6-unicast
set protocols isis interface lo0.0
```

### Device R5

```
set interfaces so-1/2/0 unit 0 family inet address 10.0.0.14/30
set interfaces so-1/2/0 unit 0 family iso
```

```

set interfaces so-1/2/1 unit 0 family inet address 10.0.0.22/30
set interfaces so-1/2/1 unit 0 family iso
set interfaces so-1/2/2 unit 0 family inet address 10.0.0.26/30
set interfaces so-1/2/2 unit 0 family iso
set interfaces so-1/2/3 unit 0 family iso
set interfaces so-1/2/3 unit 0 family inet6 address 2001:db8:0:3::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.5/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0005.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::5/128
set protocols isis topologies ipv6-unicast
set protocols isis interface so-1/2/0.0 no-ipv6-unicast
set protocols isis interface so-1/2/1.0 no-ipv6-unicast
set protocols isis interface so-1/2/2.0 no-ipv6-unicast
set protocols isis interface so-1/2/3.0
set protocols isis interface lo0.0

```

## Device R6

```

set interfaces so-1/2/0 unit 0 family inet address 10.0.0.6/30
set interfaces so-1/2/0 unit 0 family iso
set interfaces so-1/2/0 unit 0 family inet6 address 2001:db8:0:2::/64 eui-64
set interfaces so-1/2/1 unit 0 family iso
set interfaces so-1/2/1 unit 0 family inet6 address 2001:db8:0:3::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.6/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0006.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::6/128
set protocols isis topologies ipv6-unicast
set protocols isis interface so-1/2/0.0
set protocols isis interface so-1/2/1.0
set protocols isis interface lo0.0

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure an alternate IPv6 unicast topology:

### 1. Configure the interfaces.

```
[edit interfaces]
user@R1# set so-1/2/0 unit 0 family inet address 10.0.0.1/30
user@R1# set so-1/2/0 unit 0 family iso
user@R1# set so-1/2/0 unit 0 family inet6 address 2001:db8:0:5::/64 eui-64
user@R1# set so-1/2/1 unit 0 family inet address 10.0.0.13/30
user@R1# set so-1/2/1 unit 0 family iso
user@R1# set so-1/2/2 unit 0 family inet address 10.0.0.17/30
user@R1# set so-1/2/2 unit 0 family iso
user@R1# set so-1/2/2 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
user@R1# set lo0 unit 0 family inet address 192.168.0.1/32
user@R1# set lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
user@R1# set lo0 unit 0 family inet6 address 2001:db8::1/128
```

### 2. Enable IS-IS on the interfaces.

```
[edit protocols isis]
user@R1# set interface so-1/2/0.0
user@R1# set interface so-1/2/1.0
user@R1# set interface so-1/2/2.0
user@R1# set interface lo0.0
```

### 3. Enable multitopology routing on the IS-IS interfaces.

The `ipv6-unicast` statement enables multitopology IS-IS routing on all interfaces that have family `iso` and family `inet6` configured and are listed at the `[edit protocols isis interface]` hierarchy level.

```
[edit protocols isis]
user@R1# set topologies ipv6-unicast
```

### 4. Disable IPv6 unicast support on a given interface.

If you do not want to run multitopology IS-IS routing for IPv6 on a given interface, you can disable multitopology routing by including the `no-ipv6-unicast` statement in the IS-IS interface configuration.

```
[edit protocols isis]
user@R1# set interface so-1/2/1.0 no-ipv6-unicast
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces` and `show protocols` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces
so-1/2/0 {
  unit 0 {
    family inet {
      address 10.0.0.1/30;
    }
    family iso;
    family inet6 {
      address 2001:db8:0:5::/64 {
        eui-64;
      }
    }
  }
}
so-1/2/1 {
  unit 0 {
    family inet {
      address 10.0.0.13/30;
    }
    family iso;
  }
}
so-1/2/2 {
  unit 0 {
    family inet {
      address 10.0.0.17/30;
    }
    family iso;
    family inet6 {
      address 2001:db8:0:1::/64 {
        eui-64;
      }
    }
  }
}
lo0 {
```



```

unit 0 {
    family inet {
        address 192.168.0.1/32;
    }
    family iso {
        address 49.0002.0192.0168.0001.00;
    }
    family inet6 {
        address 2001:db8::1/128;
    }
}
}

```

```

user@R1# show protocols
isis {
    topologies ipv6-unicast;
    interface so-1/2/0.0;
    interface so-1/2/1.0 {
        no-ipv6-unicast;
    }
    interface so-1/2/2.0;
    interface lo0.0;
}

```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Checking the Topologies on Neighbors | 332](#)
- [Checking the IS-IS SPF Calculations | 333](#)
- [Checking the Tcpdump Output | 334](#)

Confirm that the configuration is working properly.

## Checking the Topologies on Neighbors

### Purpose

Determine what topologies are supported on neighboring IS-IS devices.

### Action

From operational mode, enter the `show isis adjacency detail` command.

```
user@R1> show isis adjacency detail
```

R2

```
Interface: so-1/2/0.0, Level: 3, State: Up, Expires in 24 secs
Priority: 0, Up/Down transitions: 1, Last transition: 05:28:16 ago
Circuit type: 3, Speaks: IP, IPv6
Topologies: Unicast, IPV6-Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 10.0.0.2
IPv6 addresses: fe80::2a0:a514:0:24c
```

R5

```
Interface: so-1/2/1.0, Level: 3, State: Up, Expires in 21 secs
Priority: 0, Up/Down transitions: 1, Last transition: 05:27:47 ago
Circuit type: 3, Speaks: IP, IPv6
Topologies: Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 10.0.0.14
```

R3

```
Interface: so-1/2/2.0, Level: 3, State: Up, Expires in 22 secs
Priority: 0, Up/Down transitions: 1, Last transition: 05:27:25 ago
Circuit type: 3, Speaks: IP, IPv6
Topologies: Unicast, IPV6-Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 10.0.0.18
IPv6 addresses: fe80::2a0:a514:0:124c
```

## Meaning

As expected, the adjacency with Device R5 only supports the IPv4 unicast topology, while the adjacencies with Device R2 and Device R3 support both the IPv4 and IPv6 topologies.

## Checking the IS-IS SPF Calculations

### Purpose

Verify that separate SPF calculations are being run for IPv4 and IPv6.

### Action

From operational mode, enter the `show isis spf brief` command.

```
user@R1> show isis spf brief
```

#### IPv4 Unicast IS-IS level 1 SPF results:

Node	Metric	Interface	NH	Via	SNPA
R6.00	20	so-1/2/1.0	IPV4 R5		
R4.00	20	so-1/2/0.0	IPV4 R2		
R5.00	10	so-1/2/1.0	IPV4 R5		
R3.00	10	so-1/2/2.0	IPV4 R3		
R2.00	10	so-1/2/0.0	IPV4 R2		
R1.00	0				

6 nodes

#### IPv4 Unicast IS-IS level 2 SPF results:

Node	Metric	Interface	NH	Via	SNPA
R6.00	20	so-1/2/1.0	IPV4 R5		
R4.00	20	so-1/2/0.0	IPV4 R2		
R5.00	10	so-1/2/1.0	IPV4 R5		
R3.00	10	so-1/2/2.0	IPV4 R3		
R2.00	10	so-1/2/0.0	IPV4 R2		
R1.00	0				

6 nodes

#### IPv6 Unicast IS-IS level 1 SPF results:

Node	Metric	Interface	NH	Via	SNPA
R5.00	40	so-1/2/0.0	IPV6 R2		
R6.00	30	so-1/2/0.0	IPV6 R2		
R4.00	20	so-1/2/0.0	IPV6 R2		
R3.00	10	so-1/2/2.0	IPV6 R3		

```
R2.00      10      so-1/2/0.0      IPV6 R2
R1.00      0
6 nodes
```

#### IPV6 Unicast IS-IS level 2 SPF results:

Node	Metric	Interface	NH	Via	SNPA
R5.00	40	so-1/2/0.0	IPV6 R2		
R6.00	30	so-1/2/0.0	IPV6 R2		
R4.00	20	so-1/2/0.0	IPV6 R2		
R3.00	10	so-1/2/2.0	IPV6 R3		
R2.00	10	so-1/2/0.0	IPV6 R2		
R1.00	0				
6 nodes					

## Meaning

As expected, SPF calculations are being performed for IPv4 and IPv6 topologies.

## Checking the Tcpdump Output

### Purpose

Verify that the link can be a member of both the IPv4 unicast topology and the IPv6 unicast topology.

### Action

```
user@R1> monitor traffic detail interface so-1/2/0.0
[...]

15:52:35.719540 In IS-IS, length 82
  p2p IIH, hlen: 20, v: 1, pdu-v: 1, sys-id-len: 6 (0), max-area: 3 (0)
  source-id: 0192.0168.0002, holding time: 27s, Flags: [Level 1, Level 2]
  circuit-id: 0x01, PDU length: 82
    Point-to-point Adjacency State TLV #240, length: 15
      Adjacency State: Up (0)
      Extended Local circuit-ID: 0x00000054
      Neighbor System-ID: 0192.0168.0001
      Neighbor Extended Local circuit-ID: 0x00000043
    Protocols supported TLV #129, length: 2
      NLPID(s): IPv4 (0xcc), IPv6 (0x8e)
    IPv4 Interface address(es) TLV #132, length: 4
```

```

IPv4 interface address: 10.0.0.2
IPv6 Interface address(es) TLV #232, length: 16
IPv6 interface address: fe80::2a0:a514:0:24c
Area address(es) TLV #1, length: 4
Area address (length: 3): 49.0002
Restart Signaling TLV #211, length: 3
Flags [none], Remaining holding time 0s
Multi Topology TLV #229, length: 4
  IPv4 unicast Topology (0x000), Flags: [none]
  IPv6 unicast Topology (0x002), Flags: [none]

```

## Meaning

The IS-IS hello (IIH) packet shows that IPv4 and IPv6 are supported. The hello packet lists valid IPv4 and IPv6 addresses, and therefore the routing device can create valid next-hop entries. The supported protocols are listed in the multitopology TLV #229.

## RELATED DOCUMENTATION

| [Example: Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses](#) | 313

## Configuring IS-IS Link and Node Link Protection

### IN THIS CHAPTER

- [Understanding Loop-Free Alternate Routes for IS-IS | 336](#)
- [Example: Configuring Node-Link Protection for IS-IS Routes in a Layer 3 VPN | 341](#)
- [Understanding Remote LFA over LDP Tunnels in IS-IS Networks | 357](#)
- [Configuring Remote LFA Backup over LDP Tunnels in an IS-IS Network | 359](#)
- [Example: Configuring Remote LFA over LDP Tunnels in IS-IS Networks | 361](#)
- [Understanding Weighted ECMP Traffic Distribution on One-Hop IS-IS Neighbors | 378](#)
- [Example: Weighted ECMP Traffic Distribution on One-Hop IS-IS Neighbors | 379](#)

## Understanding Loop-Free Alternate Routes for IS-IS

### IN THIS SECTION

- [Configuring Link Protection for IS-IS | 338](#)
- [Configuring Node-Link Protection for IS-IS | 339](#)
- [Excluding an IS-IS Interface as a Backup for Protected Interfaces | 340](#)
- [Configuring RSVP Label-Switched Paths as Backup Paths for IS-IS | 340](#)
- [Using Operational Mode Commands to Monitor Protected IS-IS Routes | 341](#)

Junos OS supports IS-IS loop-free alternate routes for enabling IP fast-reroute capability for IS-IS. Junos OS precomputes loop-free backup routes for all IS-IS routes. These backup routes are preinstalled in the Packet Forwarding Engine, which performs a local repair and implements the backup path when the link for a primary next hop for a particular route is no longer available. With local repair, the Packet Forwarding Engine can correct a path failure before it receives recomputed paths from the Routing Engine. Local repair reduces the amount of time needed to reroute traffic to less than 50 milliseconds. In

contrast, global repair can take up to 800 milliseconds to compute a new route. Local repair and global repair are thus complementary. Local repair enables traffic to continue to be routed using a backup path until global repair is able to calculate a new route.

A loop-free path is one that does not forward traffic back through the routing device to reach a given destination. That is, a neighbor whose shortest path to the destination traverses the routing device is not used as a backup route to that destination. To determine loop-free alternate paths for IS-IS routes, Junos OS runs shortest-path-first (SPF) calculations on each one-hop neighbor. You can enable support for alternate loop-free routes on any IS-IS interface. Because it is common practice to enable LDP on an interface for which IS-IS is already enabled, this feature also provides support for LDP label-switched paths (LSPs).



**NOTE:** If you enable support for alternate loop-free routes on an interface configured for both LDP and IS-IS, you can use the `traceroute` command to trace the active path to the primary next hop.

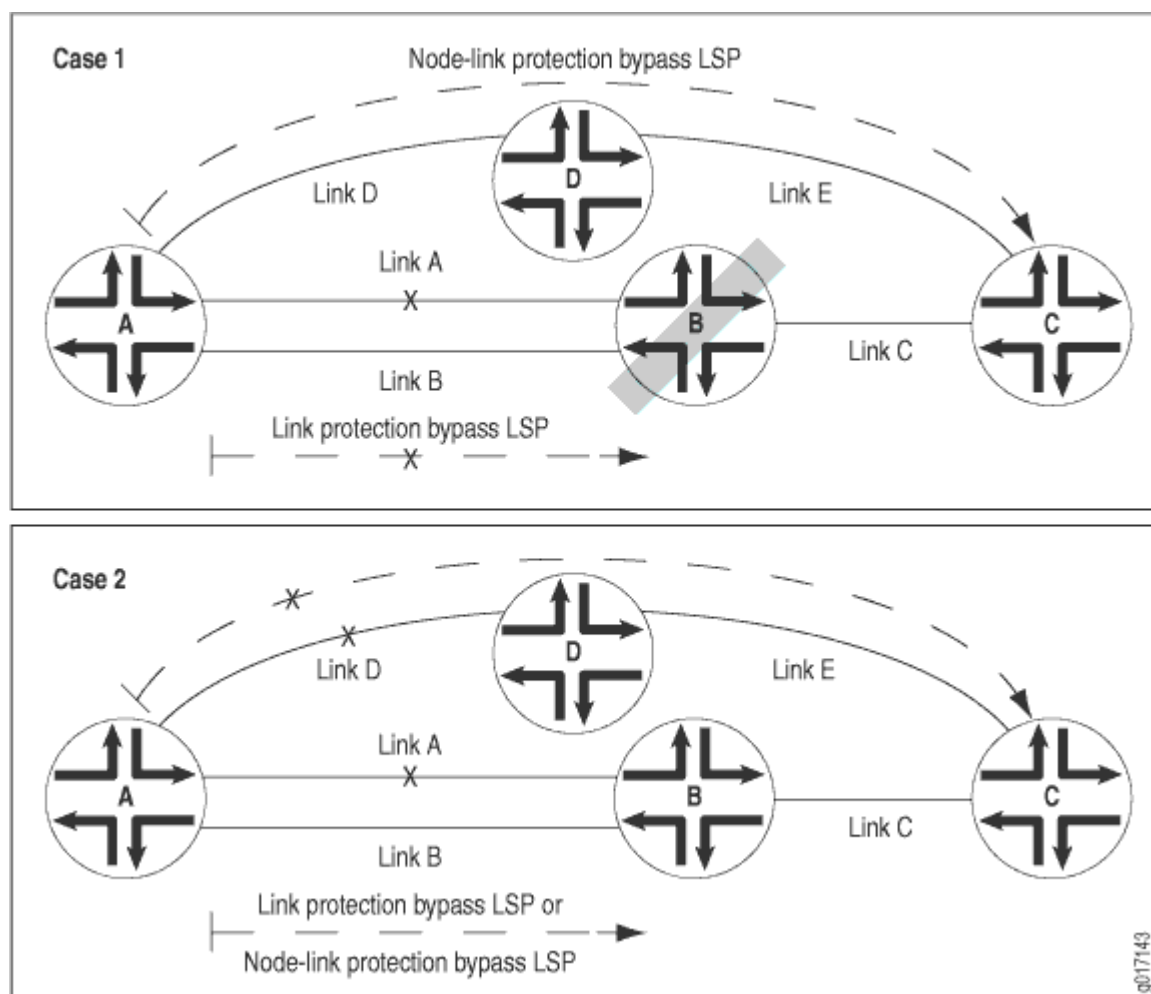
The level of backup coverage available through IS-IS routes depends on the actual network topology and is typically less than 100 percent for all destinations on any given routing device. You can extend backup coverage to include RSVP LSPs.

Junos OS provides two mechanisms for route redundancy for IS-IS through alternate loop-free routes: link protection and node-link protection. When you enable link protection or node-link protection on an IS-IS interface, Junos OS creates a single alternate path to the primary next hop for all destination routes that traverse a protected interface. Link protection offers per-link traffic protection. Use link protection when you assume that only a single link might become unavailable but that the neighboring node on the primary path would still be available through another interface.

Node-link protection establishes an alternate path through a different routing device altogether. Use node-link protection when you assume that access to a node is lost when a link is no longer available. As a result, Junos OS calculates a backup path that avoids the primary next-hop routing device.

In [Figure 27 on page 338](#), Case 2 shows how link protection allows source Router A to switch to Link B when the primary next hop Link A to destination Router C fails. However, if Router B fails, Link B also fails, and the protected Link A is lost. If node-link protection is enabled, Router A is able to switch to Link D on Router D and bypass the failed Router B altogether. As shown in Case 1, with node-link protection enabled, Router A has a node-link protection alternate path available through Router D to destination Router C. That means that if Router B fails, Router A can still reach Router C because the path from Router A to Link D remains available as an alternate backup path.

Figure 27: Link Protection and Node-Link Protection Comparison for IS-IS Routes



The Junos OS implementation of support for loop-free alternate paths for IS-IS routes is based on the following standards:

- RFC 5286, *Basic Specification for IP Fast-Reroute: Loop-free Alternates*
- RFC 5714, *IP Fast Reroute Framework*

## Configuring Link Protection for IS-IS

You can configure link protection on any interface for which IS-IS is enabled. When you enable link protection, Junos OS creates one alternate path to the primary next hop for all destination routes that traverse a protected interface. Link protection assumes that only a single link becomes unavailable but that the neighboring node would still be available through another interface.





**NOTE:** You must also configure a per-packet load-balancing routing policy to ensure that the routing protocol process installs all the next hops for a given route in the routing table.

To enable link protection, include the `link-protection` statement at the `[edit protocols isis interface interface-name]` hierarchy level:

```
[edit]
protocols {
  isis {
    interface interface-name {
      link-protection;
    }
  }
}
```

## Configuring Node-Link Protection for IS-IS

You can configure node-link protection on any interface for which IS-IS is enabled. Node-link protection establishes an alternate path through a different routing device altogether for all destination routes that traverse a protected interface. Node-link protection assumes that the entire routing device, or node, has failed. Junos OS therefore calculates a backup path that avoids the primary next-hop routing device.



**NOTE:** You must also configure a per-packet load-balancing routing policy to ensure that the routing protocol process installs all the next hops for a given route in the routing table.

To enable node-link protection, include the `node-link-protection` statement at the `[edit protocols isis interface interface-name]` hierarchy level:

```
[edit]
protocols {
  isis {
    interface interface-name {
      node-link-protection;
    }
  }
}
```

```
    }
}
```

## Excluding an IS-IS Interface as a Backup for Protected Interfaces

By default, all IS-IS interfaces that belong to the master instance or a specific routing instance are eligible as backup interfaces for protected interfaces. You can specify that any IS-IS interface be excluded from functioning as a backup interface to protected interfaces. To exclude an IS-IS interface as a backup interface, include the `no-eligible-backup` statement at the `[edit protocols isis interface interface-name]` hierarchy level:

```
[edit]
protocols {
  isis {
    interface interface-name {
      no-eligible-backup;
    }
  }
}
```

## Configuring RSVP Label-Switched Paths as Backup Paths for IS-IS

Relying on the shortest-path-first (SPF) calculation of backup paths for one-hop neighbors might result in less than 100 percent backup coverage for a specific network topology. You can enhance coverage of IS-IS and LDP label-switched paths (LSPs) by configuring RSVP LSPs as backup paths. To configure a specific RSVP LSP as a backup path, include the `backup` statement at the `[edit protocols mpls label-switched-path lsp-name]` hierarchy level:

```
[edit]
protocols {
  mpls {
    label-switched-path lsp-name {
      backup;
      to ip-address;
    }
  }
}
```

When configuring an LSP, you must specify the IP address of the egress routing device with the `to` statement. For detailed information about configuring LSPs and RSVP, see the [Junos OS MPLS Applications Library for Routing Devices](#).

## Using Operational Mode Commands to Monitor Protected IS-IS Routes

You can issue operational mode commands that provide more details about your link-protected and node-link-protected IS-IS routes. The following guidelines explain the type of information available from the output of each command:

- `show isis backup label-switched-path`—Displays which MPLS LSPs have been designated as backup paths and the current status of those LSPs.
- `show isis backup spf results`—Displays SPF calculations for each neighbor for a given destination. Indicates whether a specific interface or node has been designated as a backup path and why. Use the `no-coverage` option to display only those nodes that do not have backup coverage.
- `show isis backup coverage` —Displays the percentage of nodes and prefixes for each type of address family that is protected.
- `show isis interface detail`—Displays the type of protection (link or node-link) applied to each protected interface.

### RELATED DOCUMENTATION

| [Example: Configuring Node-Link Protection for IS-IS Routes in a Layer 3 VPN](#) | 341

## Example: Configuring Node-Link Protection for IS-IS Routes in a Layer 3 VPN

### IN THIS SECTION

- [Requirements](#) | 342
- [Overview](#) | 342
- [Configuration](#) | 343
- [Verification](#) | 352

Node-link protection establishes an alternate path through a different routing device. Use node-link protection when you assume that access to a node is lost when a link is no longer available. Junos OS calculates a backup path that avoids the primary next-hop routing device.

# Requirements

No special configuration beyond device initialization is required before configuring this example.

# Overview

## IN THIS SECTION

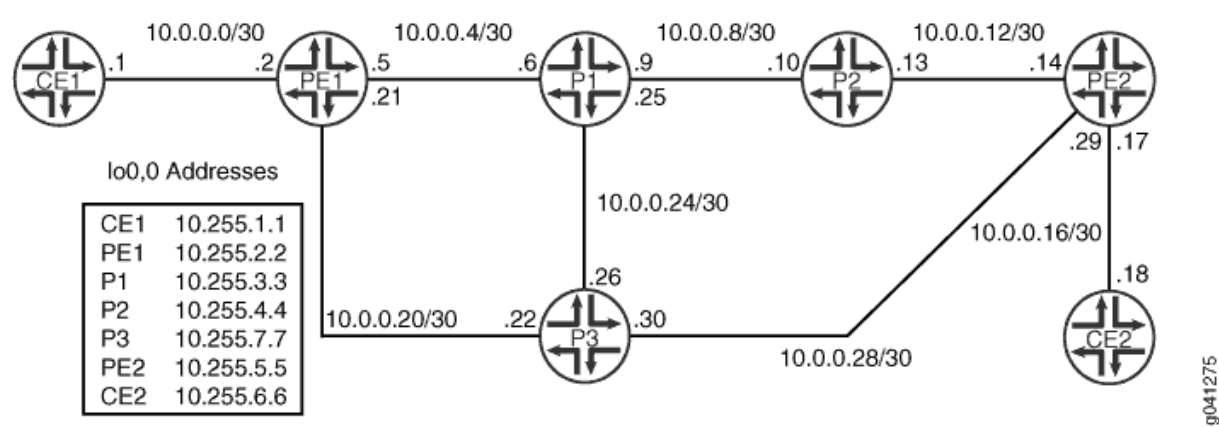
- [Topology | 343](#)

In this example, core-facing interfaces are enabled for IS-IS Level 2, LDP, and RSVP. Node-link protection is enabled on all the core-facing interfaces, which means that if the primary next hop for any destination that traverses the interfaces becomes unavailable, Junos OS uses a backup link that avoids the next-hop router altogether if necessary.

You also need to configure a routing policy that requires all traffic to use per-packet load balancing in order to enable Packet Forwarding Engine local repair. With local repair, the Packet Forwarding Engine can correct a path failure and implement a backup loop-free alternate route before it receives recomputed paths from the Routing Engine.

[Figure 28 on page 342](#) shows the topology used in this example.

**Figure 28: IS-IS Node-Link Protection Topology**



On Device PE1, an RSVP LSP is configured as a backup path for IS-IS. Relying on the shortest-path-first (SPF) calculation of backup paths for one-hop neighbors might result in less than 100 percent backup coverage for a specific network topology. You can enhance coverage of IS-IS and LDP LSPs by configuring RSVP LSPs as backup paths. To configure a specific RSVP LSP as a backup path, include the backup statement at the [edit protocols mpls label-switched-path *lsp-name*] hierarchy level.

"CLI Quick Configuration" on page 343 shows the configuration for all of the devices in Figure 28 on page 342. The section "No Link Title" on page 348 describes the steps on Device P1.

## Topology

## Configuration

### IN THIS SECTION

- Procedure | 343

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device CE1

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.1/30
set interfaces lo0 unit 0 family inet address 10.255.1.1/32
```

#### Device PE1

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.2/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.5/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
```

```

set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.21/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces fe-1/2/2 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.2.2/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0000.0202.00
set protocols rsvp interface fe-1/2/2.0
set protocols rsvp interface fe-1/2/1.0
set protocols rsvp interface lo0.0
set protocols rsvp interface fxp0.0 disable
set protocols mpls label-switched-path to-p2 backup
set protocols mpls label-switched-path to-p2 to 10.255.4.4
set protocols mpls label-switched-path to-p2 ldp-tunneling
set protocols mpls interface fe-1/2/2.0
set protocols mpls interface fe-1/2/1.0
set protocols mpls interface lo0.0
set protocols mpls interface fxp0.0 disable
set protocols bgp group l3vpn type internal
set protocols bgp group l3vpn local-address 10.255.2.2
set protocols bgp group l3vpn family inet-vpn unicast
set protocols bgp group l3vpn peer-as 65534
set protocols bgp group l3vpn local-as 65534
set protocols bgp group l3vpn neighbor 10.255.5.5
set protocols isis spf-options delay 1000
set protocols isis interface all node-link-protection
set protocols isis interface all level 2 metric 10
set protocols isis interface all level 1 disable
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 level 2 metric 0
set protocols ldp deaggregate
set protocols ldp interface fe-1/2/1.0
set protocols ldp interface fe-1/2/2.0
set protocols ldp interface fxp0.0 disable
set protocols ldp interface lo0.0
set policy-options policy-statement ecmp term 1 then load-balance per-packet
set routing-instances VPN-A instance-type vrf
set routing-instances VPN-A interface fe-1/2/0.0
set routing-instances VPN-A route-distinguisher 65534:1234
set routing-instances VPN-A vrf-target target:65534:1234
set routing-instances VPN-A routing-options static route 10.255.1.1/32 next-hop 10.0.0.1
set routing-options autonomous-system 65534
set routing-options forwarding-table export ecmp

```

## Device P1

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.6/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.9/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.25/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces fe-1/2/2 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.3.3/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0000.0303.00
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols isis spf-options delay 1000
set protocols isis interface all node-link-protection
set protocols isis interface all level 2 metric 10
set protocols isis interface all level 1 disable
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 level 2 metric 0
set protocols ldp deaggregate
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set policy-options policy-statement ecmp term 1 then load-balance per-packet
set routing-options forwarding-table export ecmp

```

## Device P2

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.10/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.13/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.4.4/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0000.0404.00
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable

```

```

set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols isis spf-options delay 1000
set protocols isis interface all node-link-protection
set protocols isis interface all level 2 metric 10
set protocols isis interface all level 1 disable
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 level 2 metric 0
set protocols ldp deaggregate
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set policy-options policy-statement ecmp term 1 then load-balance per-packet
set routing-options forwarding-table export ecmp

```

### Device P3

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.22/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.26/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.30/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces fe-1/2/2 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.7.7/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0000.0707.00
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols isis spf-options delay 1000
set protocols isis interface all node-link-protection
set protocols isis interface all level 2 metric 10
set protocols isis interface all level 1 disable
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 level 2 metric 0
set protocols ldp deaggregate
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable

```



```
set policy-options policy-statement ecmp term 1 then load-balance per-packet
set routing-options forwarding-table export ecmp
```

## Device PE2

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.14/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.17/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.29/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces fe-1/2/2 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.5.5/32
set interfaces lo0 unit 0 family iso address 49.0001.0010.0000.0505.00
set protocols rsvp interface fe-1/2/0.0
set protocols rsvp interface fe-1/2/2.0
set protocols rsvp interface lo0.0
set protocols rsvp interface fxp0.0 disable
set protocols mpls interface fe-1/2/0.0
set protocols mpls interface fe-1/2/2.0
set protocols mpls interface lo0.0
set protocols mpls interface fxp0.0 disable
set protocols bgp group l3vpn type internal
set protocols bgp group l3vpn local-address 10.255.5.5
set protocols bgp group l3vpn family inet-vpn unicast
set protocols bgp group l3vpn peer-as 65534
set protocols bgp group l3vpn local-as 65534
set protocols bgp group l3vpn neighbor 10.255.2.2
set protocols isis spf-options delay 1000
set protocols isis interface all node-link-protection
set protocols isis interface all level 2 metric 10
set protocols isis interface all level 1 disable
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 level 2 metric 0
set protocols ldp deaggregate
set protocols ldp interface fe-1/2/0.0
set protocols ldp interface fe-1/2/2.0
set protocols ldp interface fxp0.0 disable
set protocols ldp interface lo0.0
set policy-options policy-statement ecmp term 1 then load-balance per-packet
set routing-instances VPN-A instance-type vrf
```

```

set routing-instances VPN-A interface fe-1/2/1.0
set routing-instances VPN-A route-distinguisher 65534:1234
set routing-instances VPN-A vrf-target target:65534:1234
set routing-instances VPN-A routing-options static route 10.255.1.1/32 next-hop 10.0.0.18
set routing-options autonomous-system 65534
set routing-options forwarding-table export ecmp

```

## Device CE2

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.18/30
set interfaces lo0 unit 0 family inet address 10.255.6.6/32

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure multi-level IS-IS:

### 1. Configure the interfaces.

Enable IS-IS and MPLS.

```

[edit interfaces]
user@P1# set fe-1/2/0 unit 0 family inet address 10.0.0.6/30
user@P1# set fe-1/2/0 unit 0 family iso
user@P1# set fe-1/2/0 unit 0 family mpls
user@P1# set fe-1/2/1 unit 0 family inet address 10.0.0.9/30
user@P1# set fe-1/2/1 unit 0 family iso
user@P1# set fe-1/2/1 unit 0 family mpls
user@P1# set fe-1/2/2 unit 0 family inet address 10.0.0.25/30
user@P1# set fe-1/2/2 unit 0 family iso
user@P1# set fe-1/2/2 unit 0 family mpls
user@P1# set lo0 unit 0 family inet address 10.255.3.3/32
user@P1# set lo0 unit 0 family iso address 49.0001.0010.0000.0303.00

```

### 2. Configure the IS-IS interfaces for Level 2.

```

[edit protocols]
user@P1# set isis interface all level 2 metric 10

```

```

user@P1# set isis interface all level 1 disable
user@P1# set isis interface fxp0.0 disable
user@P1# set isis interface lo0.0 level 2 metric 0

```

3. Enable IS-IS node-link protection, which also automatically extends backup coverage to all LDP LSPs.

```

[edit protocols]
user@P1# set isis interface all node-link-protection

```

4. (Optional) Configure a 1000-millisecond time interval between the detection of a topology change and when the SPF algorithm runs.

```

[edit protocols]
user@P1# set isis spf-options delay 1000

```

5. Configure MPLS to use both RSVP and LDP label-switched paths (LSPs).

```

[edit protocols]
user@P1# set mpls interface all
user@P1# set mpls interface fxp0.0 disable
user@P1# set rsvp interface all
user@P1# set rsvp interface fxp0.0 disable
user@P1# set ldp interface all
user@P1# set ldp interface fxp0.0 disable

```

6. (Optional) For LDP, enable forwarding equivalence class (FEC) deaggregation, which results in faster global convergence.

```

[edit protocols]
user@P1# set ldp deaggregate

```

7. To enable Packet Forwarding Engine local repair, establish a policy that forces the routing protocol process to install all the next hops for a given route.

This policy ensures that the backup route is installed in the forwarding table used by the Packet Forwarding Engine to forward traffic to a given destination.

```
[edit policy-options policy-statement ecmp term 1]
user@P1# set then load-balance per-packet
```

8. Apply the policy to the forwarding table of the local router with the export statement.

```
[edit routing-options forwarding-table]
user@P1# set export ecmp
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, `show policy-options`, and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@P1# show interfaces
fe-1/2/0 {
  unit 0 {
    family inet {
      address 10.0.0.6/30;
    }
    family iso;
    family mpls;
  }
}
fe-1/2/1 {
  unit 0 {
    family inet {
      address 10.0.0.9/30;
    }
    family iso;
    family mpls;
  }
}
fe-1/2/2 {
  unit 0 {
    family inet {
```

```

        address 10.0.0.25/30;
    }
    family iso;
    family mpls;
}
}
lo0 {
    unit 0 {
        family inet {
            address 10.255.3.3/32;
        }
        family iso {
            address 49.0001.0010.0000.0303.00;
        }
    }
}
}

```

```

user@P1# show protocols
rsvp {
    interface all;
    interface fxp0.0 {
        disable;
    }
}
mpls {
    interface all;
    interface fxp0.0 {
        disable;
    }
}
isis {
    spf-options delay 1000;
    interface all {
        node-link-protection;
        level 2 metric 10;
        level 1 disable;
    }
    interface fxp0.0 {
        disable;
    }
    interface lo0.0 {

```

```

        level 2 metric 0;
    }
}
ldp {
    deaggregate;
    interface all;
    interface fxp0.0 {
        disable;
    }
}

```

```

user@P1# show policy-options
policy-statement ecmp {
    term 1 {
        then {
            load-balance per-packet;
        }
    }
}

```

```

user@P1# show routing-options
forwarding-table {
    export ecmp;
}

```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Checking the MPLS LSP Backup Path | 353](#)
- [Checking Which Next-Hop Neighbors Are Designated as Backup Paths to the Destination Node | 353](#)
- [Checking the Backup Coverage | 355](#)
- [Checking the Type of Protection Configured | 356](#)

Confirm that the configuration is working properly.

## Checking the MPLS LSP Backup Path

### Purpose

Display information about the MPLS label-switched-paths (LSPs) designated as the backup route for the IS-IS routes.

### Action

On Device PE1, from operational mode, enter the `show isis backup label-switched-path` command.

```
user@PE1> show isis backup label-switched-path
Backup MPLS LSPs:
to-p2, Egress: 10.255.4.4, Status: up, Last change: 01:17:45
  TE-metric: 19, Metric: 0, Refcount: 1
```

### Meaning

The output shows that the backup path is up and operational.

## Checking Which Next-Hop Neighbors Are Designated as Backup Paths to the Destination Node

### Purpose

Display SPF calculations for each neighbor for a given destination.

### Action

On Device PE1, from operational mode, enter the `show isis backup spf results` command.

```
user@PE1> show isis backup spf results

IS-IS level 1 SPF results:
  0 nodes

IS-IS level 2 SPF results:
PE2.00
  Primary next-hop: fe-1/2/2.0, IPV4, P3, SNPA: 0:5:85:8f:c8:bd
```

Root: P2, Root Metric: 20, Metric: 10, Root Preference: 0x0  
 track-item: P2.00-00

Eligible, Backup next-hop: fe-1/2/1.0, LSP, to-p2

Root: P3, Root Metric: 10, Metric: 10, Root Preference: 0x0  
 Not eligible, Reason: Interface is already covered

Root: P1, Root Metric: 10, Metric: 20, Root Preference: 0x0  
 track-item: P3.00-00

Not eligible, Reason: Interface is already covered

P2.00

Primary next-hop: fe-1/2/1.0, IPV4, P1, SNPA: 0:5:85:8f:c8:bd

Root: P2, Root Metric: 20, Metric: 0, Root Preference: 0x0  
 track-item: P2.00-00

Not eligible, Reason: Primary next-hop link fate sharing

Root: P1, Root Metric: 10, Metric: 10, Root Preference: 0x0  
 Not eligible, Reason: Primary next-hop link fate sharing

Root: P3, Root Metric: 10, Metric: 20, Root Preference: 0x0  
 track-item: P1.00-00

Not eligible, Reason: Primary next-hop node fate sharing

P3.00

Primary next-hop: fe-1/2/2.0, IPV4, P3, SNPA: 0:5:85:8f:c8:bd

Root: P2, Root Metric: 20, Metric: 20, Root Preference: 0x0  
 track-item: P3.00-00

track-item: P2.00-00

track-item: P1.00-00

Eligible, Backup next-hop: fe-1/2/1.0, LSP, to-p2

Root: P3, Root Metric: 10, Metric: 0, Root Preference: 0x0  
 Not eligible, Reason: Interface is already covered

Root: P1, Root Metric: 10, Metric: 10, Root Preference: 0x0  
 track-item: P3.00-00

Not eligible, Reason: Interface is already covered

P1.00

Primary next-hop: fe-1/2/1.0, IPV4, P1, SNPA: 0:5:85:8f:c8:bd

Root: P2, Root Metric: 20, Metric: 10, Root Preference: 0x0  
 track-item: P2.00-00

track-item: P1.00-00

Not eligible, Reason: Primary next-hop link fate sharing

Root: P1, Root Metric: 10, Metric: 0, Root Preference: 0x0  
 Not eligible, Reason: Primary next-hop link fate sharing

Root: P3, Root Metric: 10, Metric: 10, Root Preference: 0x0  
 track-item: P1.00-00

Eligible, Backup next-hop: fe-1/2/2.0, IPV4, P3, SNPA: 0:5:85:8f:c8:bd

4 nodes



## Meaning

The output indicates whether a specific interface or node has been designated as a backup path and why.

## Checking the Backup Coverage

### Purpose

Check the percentage of protected nodes and prefixes.

### Action

From operational mode, enter the `show isis backup coverage` command.

```
user@PE1> show isis backup coverage
Backup Coverage:
Topology      Level  Node   IPv4   IPv6   CLNS
IPV4 Unicast  1      0.00%  0.00%  0.00%  0.00%
IPV4 Unicast  2      75.00% 87.50%  0.00%  0.00%
```

```
user@P1> show isis backup coverage
Backup Coverage:
Topology      Level  Node   IPv4   IPv6   CLNS
IPV4 Unicast  1      0.00%  0.00%  0.00%  0.00%
IPV4 Unicast  2      75.00% 71.43%  0.00%  0.00%
```

```
user@P2> show isis backup coverage
Backup Coverage:
Topology      Level  Node   IPv4   IPv6   CLNS
IPV4 Unicast  1      0.00%  0.00%  0.00%  0.00%
IPV4 Unicast  2      50.00% 37.50%  0.00%  0.00%
```

```
user@P3> show isis backup coverage
Backup Coverage:
Topology      Level  Node   IPv4   IPv6   CLNS
```

IPv4 Unicast	1	0.00%	0.00%	0.00%	0.00%
IPv4 Unicast	2	75.00%	71.43%	0.00%	0.00%

```

user@PE2> show isis backup coverage
Backup Coverage:
Topology      Level  Node   IPv4   IPv6   CLNS
IPV4 Unicast  1      0.00%  0.00%  0.00%  0.00%
IPV4 Unicast  2      50.00% 37.50%  0.00%  0.00%

```

## Meaning

The level of backup coverage available through IS-IS routes depends on the actual network topology and is typically less than 100 percent for all destinations on any given routing device. You can extend backup coverage to include RSVP LSPs.

## Checking the Type of Protection Configured

### Purpose

On all nodes in the IS-IS domain, check the type and percentage of protected nodes and prefixes.

### Action

From operational mode, enter the `show isis interface detail` command.

```

user@PE1> show isis interface detail

IS-IS interface database:
lo0.0
  Index: 76, State: 0x6, Circuit id: 0x1, Circuit type: 0
  LSP interval: 100 ms, CSNP interval: disabled
  Adjacency advertisement: Advertise
  Level Adjacencies Priority Metric Hello (s) Hold (s) Designated Router
    1           0       64      0 Passive
    2           0       64      0 Passive
fe-1/2/2.0
  Index: 79, State: 0x6, Circuit id: 0x1, Circuit type: 2
  LSP interval: 100 ms, CSNP interval: 10 s
  Adjacency advertisement: Advertise

```

**Protection Type: Node Link**

Level	Adjacencies	Priority	Metric	Hello (s)	Hold (s)	Designated Router
2	1	64	10	9.000	27	P3.03 (not us)

fe-1/2/1.0

Index: 77, State: 0x6, Circuit id: 0x1, Circuit type: 2

LSP interval: 100 ms, CSNP interval: 10 s

Adjacency advertisement: Advertise

**Protection Type: Node Link**

Level	Adjacencies	Priority	Metric	Hello (s)	Hold (s)	Designated Router
2	1	64	10	9.000	27	P1.02 (not us)

**Meaning**

The output shows that node-link protection is configured on the interfaces.

**RELATED DOCUMENTATION**

[Understanding Loop-Free Alternate Routes for IS-IS | 336](#)

**Understanding Remote LFA over LDP Tunnels in IS-IS Networks**

In an IS-IS network, a loop free alternate (LFA) is a directly connected neighbor that provides precomputed backup paths to the destinations reachable through the protected link on the point of local repair (PLR). A remote LFA is not directly connected to the PLR and provides precomputed backup paths using dynamically created LDP tunnels to the remote LFA node. The PLR uses this remote LFA backup path when the primary link fails. The primary goal of the remote LFA is to increase backup coverage for the IS-IS networks and provide protection for Layer 1 metro-rings.

However, LFAs do not provide full backup coverage for IS-IS based Metro Ethernet networks, which are often deployed in a ring topology. To overcome this limitation, the Resource Reservation Protocol Resource Reservation Protocol - Traffic Engineering (RSVP-TE) backup tunnels are commonly used to extend the backup coverage. However, a majority of network providers have already implemented LDP as the MPLS tunnel setup protocol and do not want to implement the RSVP-TE protocol merely for backup coverage. LDP automatically brings up transport tunnels to all potential destinations in an IS-IS network and hence is the preferred protocol. The existing LDP implemented for the MPLS tunnel setup can be reused for protection of IS-IS networks and subsequent LDP destinations, thereby eliminating the need for RSVP-TE backup tunnels for backup coverage.

To calculate the remote LFA backup path, the IS-IS protocol determines the remote LFA node in the following manner:

1. Calculates the reverse shortest path first from the adjacent router across the protected link of a PLR. The reverse shortest path first uses the incoming link metric instead of the outgoing link metric to reach a neighboring node.

The result is a set of links and nodes, which is the shortest path from each leaf node to the root node.

2. Calculates the shortest path first (SPF) on the remaining adjacent routers to find the list of nodes that can be reached without traversing the link being protected.

The result is another set of links and nodes on the shortest path from the root node to all leaf nodes.

3. Determines the common nodes from the above results, These nodes are the remote LFAs.

IS-IS listens to the advertised labels for the LDP routes. For each advertised LDP route, IS-IS checks whether it contains an LDP supplied next hop. If the corresponding IS-IS route does have a backup next hop, then IS-IS runs the backup policy and adds an additional tracking route with the corresponding LDP label-switched path next hop as the backup next hop. If there are no backup next hops, LDP builds a dynamic LDP tunnel to the remote LFA, and LDP establishes a targeted adjacency between the remote LFA node and the PLR node. This backup route has two LDP labels. The top label is the IS-IS route, which denotes the backup path from the PLR to the remote LFA route. The bottom label is the LDP MPLS label-switched path that denotes the route for reaching the ultimate destination from the remote LFA. When an LDP session goes down and a remote tunnel is no longer available, IS-IS changes all the routes that have been using this backup LDP tunnel.



**NOTE:** Currently, Junos OS supports only IPv4 transport LSPs. If you need to reuse IPv4 transport LSPs for IPv6 IGP networks, add an IPv6 explicit NULL label to the label stack of the tracking route. The system automatically converts the IPv4 LSP to an IPv6 LSP.

LDP might be vulnerable by an automatically targeted adjacency, and these threats can be mitigated using all or some of the following mechanisms:

- Remote LFAs that are several hops away use extended hello messages to indicate willingness to establish a targeted LDP session. A remote LFA can reduce the threat of spoofed extended hellos by filtering them and accepting only those originating at sources permitted by an access or filter list.
- There is a need to authenticate with TCP-MD5 all auto-targeted LDP sessions in the given IGP/LDP domain using apply groups or LDP global-level authentication.
- As an added security measure, the repair or remote tunnel endpoint routers should be assigned from a set of addresses that are not reachable from outside of the routing domain.

## RELATED DOCUMENTATION

*auto-targeted-session*

[Configuring Remote LFA Backup over LDP Tunnels in an IS-IS Network | 359](#)

[Example: Configuring Remote LFA over LDP Tunnels in IS-IS Networks | 361](#)

## Configuring Remote LFA Backup over LDP Tunnels in an IS-IS Network

The primary goal of a remote loop-free alternate (LFA) is to increase backup coverage for IS-IS routes and provide protection especially for Layer 1 metro-rings. The existing LDP implemented for the MPLS tunnel setup can be reused for protection of IS-IS networks and subsequent LDP destinations. The IS-IS protocol creates a dynamic LDP tunnel to reach the remote LFA node from the point of local repair (PLR). The PLR uses this remote LFA backup path when the primary link fails.

Before you configure remote LFA over LDP tunnels in an IS-IS network, you must do the following:

1. Enable LDP on the loopback interface.

Configure a loopback interface because an LDP targeted adjacency cannot be formed without a loopback interface. LDP targeted adjacency is essential for determining remote LFA backup paths.

2. Make sure that remote LFA allows asymmetric remote neighbor discovery—that is, it must send periodic targeted hello messages to the router that initiated the remote neighbor for LDP auto-targeted adjacency.
3. Configure link protection or node-link protection on the PLR.

To configure remote LFA backup over LDP tunnels in an IS-IS network:

1. Enable remote LFA backup to determine the backup next hop using dynamic LDP label-switched path.

```
[edit protocols isis backup-spf-options]
user@host# set remote-backup-calculation
```

2. (Optional) Include the node-link-degradation statement even if node-link protection is not configured for a given interface.

The device uses the configured link protection LFA as the backup for the primary link.

```
[edit protocols isis backup-spf-options]
user@host# set node-link-degradation
```

3. Enable automatically targeted LDP sessions using the loopback addresses between the PLR and the remote LFA node.

```
[edit protocols ldp]
user@host# set auto-targeted-session
```

4. Specify a time interval for which the targeted LDP sessions are kept up even after the remote LFA node goes down.

```
[edit protocols ldp auto-targeted-session]
user@host# set teardown-delay seconds
```

For example, to set a teardown delay value of 60 seconds:

```
[edit protocols ldp auto-targeted-session]
user@host# set teardown-delay 60
```

5. Specify the maximum number of automatically targeted LDP sessions to optimize memory usage.

```
[edit protocols ldp auto-targeted-session]
user@host# set maximum-sessions number of sessions
```

For example, to set a maximum sessions allowed to 20:

```
[edit protocols ldp auto-targeted-session]
user@host# set maximum-sessions 20
```

## RELATED DOCUMENTATION

*auto-targeted-session*

---

[Example: Configuring Remote LFA over LDP Tunnels in IS-IS Networks | 361](#)

---

[Understanding Remote LFA over LDP Tunnels in IS-IS Networks | 357](#)

## Example: Configuring Remote LFA over LDP Tunnels in IS-IS Networks

### IN THIS SECTION

- [Requirements | 361](#)
- [Overview | 361](#)
- [Configuration | 364](#)
- [Verification | 373](#)

This example shows how to configure remote LFA for LDP tunnels in an IS-IS network for extending backup protection.

### Requirements

This example uses the following hardware and software components:

- Six routers with IS-IS protocol and LDP enabled on the connected interfaces.
- Junos OS running on all devices.

Before you configure remote LFA over LDP tunnels in IS-IS networks, make sure of the following:

- LDP is enabled on the loopback interface. Without a loopback interface, LDP targeted adjacency cannot be formed. Remote LFA cannot be configured without LDP targeted adjacency.
- Remote LFA must allow asymmetric remote neighbor discovery, that is, it must send periodic targeted hellos to the router that initiated the remote neighbor for LDP auto targeted adjacency.
- Link protection or node-link protection must be configured on the point of local repair (PLR).

### Overview

#### IN THIS SECTION

- [Topology | 362](#)

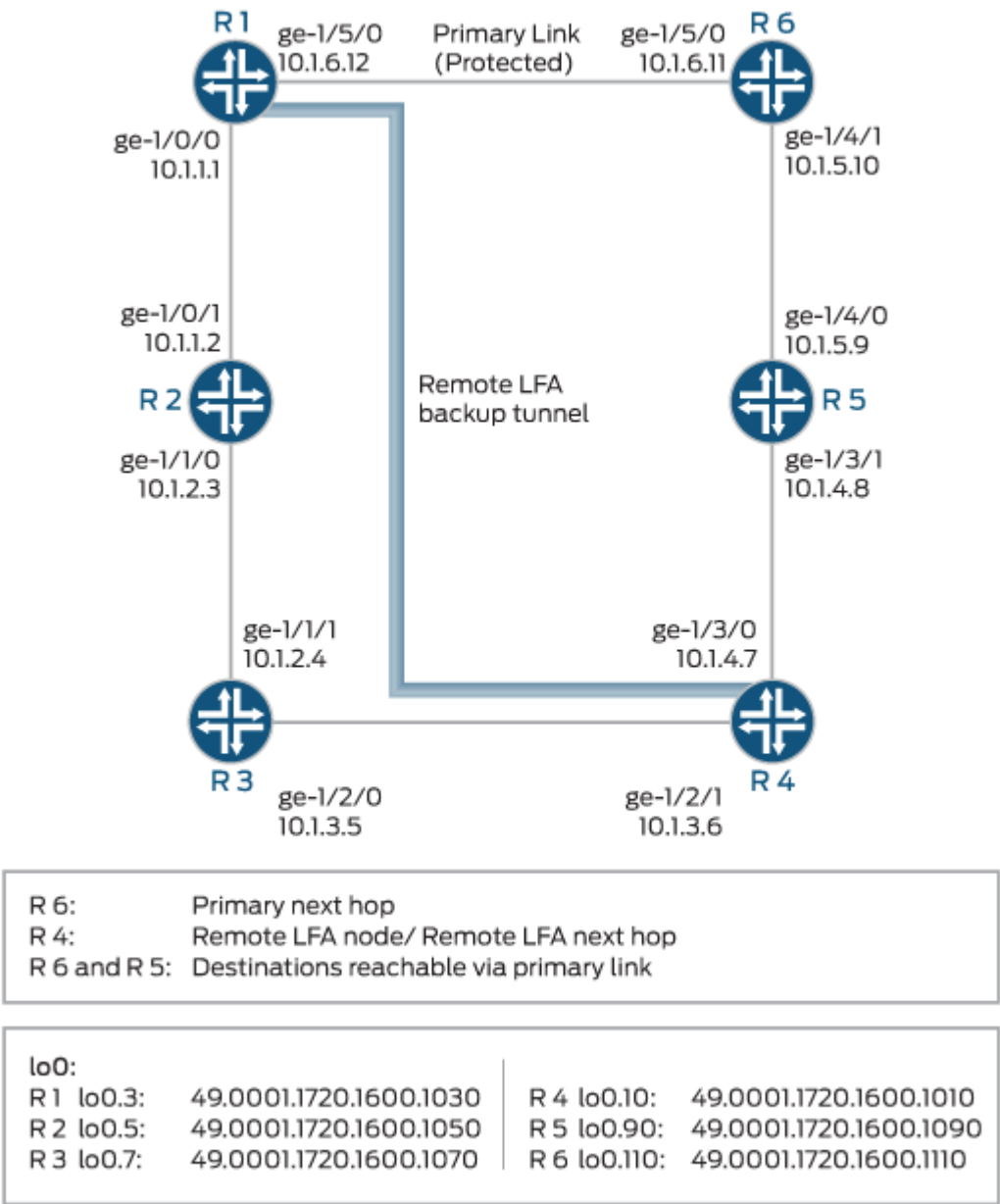
The example includes six routers in a ring topology. Configure the IS-IS protocol on the directly connected interfaces. Device R1 is the PLR. This example verifies that Junos OS updates the routing table of Device R1 with LDP next-hop routes as the backup route.

### Topology

[Figure 29 on page 363](#) shows the topology used in this example.



Figure 29: Configuring Remote LFA over LDP Tunnels in IS-IS Networks



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 364](#)
- [Configuring Device R1 | 368](#)
- [Results | 371](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

#### Router R1

```
set interfaces ge-1/0/0 unit 1 description R1->R2
set interfaces ge-1/0/0 unit 1 family inet address 10.1.1.1/24
set interfaces ge-1/0/0 unit 1 family iso
set interfaces ge-1/0/0 unit 1 family mpls
set interfaces ge-1/5/0 unit 12 description R1->R6
set interfaces ge-1/5/0 unit 12 family inet address 10.1.6.12/24
set interfaces ge-1/5/0 unit 12 family iso
set interfaces ge-1/5/0 unit 12 family mpls
set interfaces lo0 unit 3 family inet address 10.255.102.128/32
set interfaces lo0 unit 3 family iso address 49.0001.1720.1600.1030.00
set protocols isis interface ge-1/0/0.1
set protocols isis interface ge-1/5/0.12 link-protection
set protocols isis interface lo0.12 passive
set protocols isis interface all level 2 metric 10
set protocols isis interface fxp0.0 disable
set protocols isis spf-options delay 1000
set protocols isis interface all node-link-protection
set protocols isis backup-spf-options remote-backup-calculation
set protocols isis backup-spf-options node-link-degradation
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
```

```

set protocols ldp auto-targeted-session
set protocols ldp auto-targeted-session teardown-delay 60
set protocols ldp auto-targeted-session maximum-sessions 20
set protocols ldp deaggregate
set policy-options policy-statement ecmp term 1 then load-balance per-packet
set routing-options forwarding-table export ecmp

```

## Router R2

```

set interfaces ge-1/0/1 unit 2 description R2>R1
set interfaces ge-1/0/1 unit 2 family inet address 10.1.1.2/24
set interfaces ge-1/0/1 unit 2 family iso
set interfaces ge-1/0/1 unit 2 family mpls
set interfaces ge-1/1/0 unit 3 description R2->R3
set interfaces ge-1/1/0 unit 3 family inet address 10.1.2.3/24
set interfaces ge-1/1/0 unit 3 family iso
set interfaces ge-1/1/0 unit 3 family mpls
set interfaces lo0 unit 5 family inet address 10.255.102.178/32
set interfaces lo0 unit 5 family iso address 49.0001.1720.1600.1050.00
set protocols isis interface ge-1/0/1.2
set protocols isis interface ge-1/1/0.3
set protocols isis interface lo0.3 passive
set protocols isis interface all level 2 metric 10
set protocols isis interface fxp0.0 disable
set protocols isis spf-options delay 1000
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set protocols ldp auto-targeted-session
set protocols ldp auto-targeted-session teardown-delay 60
set protocols ldp auto-targeted-session maximum-sessions 20
set protocols ldp deaggregate

```

## Router R3

```

set interfaces ge-1/1/1 unit 4 description R3->R2
set interfaces ge-1/1/1 unit 4 family inet address 10.1.2.4/24
set interfaces ge-1/1/1 unit 4 family iso
set interfaces ge-1/1/1 unit 4 family mpls
set interfaces ge-1/2/0 unit 5 description R3->R4

```

```

set interfaces ge-1/2/0 unit 5 family inet address 10.1.3.5/24
set interfaces ge-1/2/0 unit 5 family iso
set interfaces ge-1/2/0 unit 5 family mpls
set interfaces lo0 unit 7 family inet address 10.255.102.146/32
set interfaces lo0 unit 7 family iso address 49.0001.1720.1600.1070.00
set protocols isis interface ge-1/1/1.4
set protocols isis interface ge-1/2/0.5
set protocols isis interface lo0.5 passive
set protocols isis interface all level 2 metric 10
set protocols isis interface fxp0.0 disable
set protocols isis spf-options delay 1000
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set protocols ldp auto-targeted-session
set protocols ldp auto-targeted-session teardown-delay 60
set protocols ldp auto-targeted-session maximum-sessions 20
set protocols ldp deaggregate

```

#### Router R4

```

set interfaces ge-1/2/1 unit 6 description R4->R3
set interfaces ge-1/2/1 unit 6 family inet address 10.1.3.6/24
set interfaces ge-1/2/1 unit 6 family iso
set interfaces ge-1/2/1 unit 6 family mpls
set interfaces ge-1/3/0 unit 7 description R4->R5
set interfaces ge-1/3/0 unit 7 family inet address 10.1.4.7/24
set interfaces ge-1/3/0 unit 7 family iso
set interfaces ge-1/3/0 unit 7 family mpls
set interfaces lo0 unit 10 family inet address 10.255.102.156/32
set interfaces lo0 unit 10 family iso address 49.0001.1720.1600.1010.00
set protocols isis interface ge-1/2/1.6
set protocols isis interface ge-1/3/0.7
set protocols isis interface lo0.7 passive
set protocols isis interface all level 2 metric 10
set protocols isis interface fxp0.0 disable
set protocols isis spf-options delay 1000
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable

```

```

set protocols ldp auto-targeted-session
set protocols ldp auto-targeted-session teardown-delay 60
set protocols ldp auto-targeted-session maximum-sessions 20
set protocols ldp deaggregate

```

## Router R5

```

set interfaces ge-1/3/1 unit 8 description R5->R4
set interfaces ge-1/3/1 unit 8 family inet address 10.1.4.8/24
set interfaces ge-1/3/1 unit 8 family iso
set interfaces ge-1/3/1 unit 8 family mpls
set interfaces ge-1/4/0 unit 9 description R5->R6
set interfaces ge-1/4/0 unit 9 family inet address 10.1.5.9/24
set interfaces ge-1/4/0 unit 9 family iso
set interfaces ge-1/4/0 unit 9 family mpls
set interfaces lo0 unit 90 family inet address 10.255.102.166/32
set interfaces lo0 unit 90 family iso address 49.0001.1720.1600.1090.00
set protocols isis interface ge-1/3/1.8
set protocols isis interface ge-1/4/0.9
set protocols isis interface lo0.9 passive
set protocols isis interface all level 2 metric 10
set protocols isis interface fxp0.0 disable
set protocols isis spf-options delay 1000
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set ldp auto-targeted-session
set ldp auto-targeted-session teardown-delay 60
set ldp auto-targeted-session maximum-sessions 20
set protocols ldp deaggregate

```

## Router R6

```

set interfaces ge-1/4/1 unit 10 description R6->R5
set interfaces ge-1/4/1 unit 10 family inet address 10.1.5.10/24
set interfaces ge-1/4/1 unit 10 family iso
set interfaces ge-1/4/1 unit 10 family mpls
set interfaces ge-1/5/0 unit 11 description R6->R1
set interfaces ge-1/5/0 unit 11 family inet address 10.1.6.11/24
set interfaces ge-1/5/0 unit 11 family iso

```

```

set interfaces ge-1/5/0 unit 11 family mpls
set interfaces lo0 unit 110 family inet address 10.255.102.136/32

set interfaces lo0 unit 110 family iso address 49.0001.1720.1600.1110.00
set protocols isis interface ge-1/4/1.10
set protocols isis interface ge-1/5/0.11
set protocols isis interface lo0.11 passive
set protocols isis interface all level 2 metric 10
set protocols isis interface fxp0.0 disable
set protocols isis spf-options delay 1000
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set protocols ldp auto-targeted-session teardown-delay 60
set protocols ldp auto-targeted-session maximum-sessions 20
set protocols ldp deaggregate

```

## Configuring Device R1

### Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).



**NOTE:** Repeat this procedure except Step 4 and 5 for every Juniper Networks router in the IGP domain, modifying the appropriate interface names, addresses, and any other parameters.

To configure Device R1:

1. Configure the interfaces.

```

[edit interfaces]
user@R1# set ge-1/0/0 unit 1 description R1->R2
user@R1# set ge-1/0/0 unit 1 family inet address 10.1.1.1/24
user@R1# set ge-1/0/0 unit 1 family iso
user@R1# set ge-1/0/0 unit 1 family mpls
user@R1# set ge-1/5/0 unit 12 description R1->R6
user@R1# set ge-1/5/0 unit 12 family inet address 10.1.6.12/24

```

```

user@R1# set ge-1/5/0 unit 12 family iso
user@R1# set ge-1/5/0 unit 12 family mpls

```

2. Assign a loopback address to the device.

```

[edit interfaces lo0 unit 3]
user@R1# set family inet address 10.255.102.128/32
user@R1# set family iso address 49.0001.1720.1600.1030.00

```

3. Configure the IS-IS interface for level 2 and the metric value on all the interfaces, and enable link protection on the protected interface.

```

[edit protocols isis]
user@R1# set interface all level 2 metric 10
user@R1# set interface lo0.12 passive
user@R1# set interface fxp0.0 disable
user@R1# set interface ge-1/0/0.1
user@R1# set interface ge-1/5/0.12 link-protection

```

4. Enable IS-IS node-link protection, which also automatically extends backup coverage to all LDP label-switched paths.

```

[edit protocols isis]
user@R1# set spf-options delay 1000
user@R1# set interface all node-link-protection

```

5. Enable remote LFA backup which calculates the backup next hop using dynamic LDP label-switched path.

(Optional) When you include the node link degradation statement even if node protection LFA is not configured for a given destination, the device uses the configured link protection LFA as the backup for the primary link.

```

[edit protocols isis]
user@R1# set backup-spf-options remote-backup-calculation
user@R1# set backup-spf-options node-link-degradation

```

6. Configure MPLS to use LDP label-switched paths for all interfaces on the device.

```
[edit protocols]
user@R1# set mpls interface all
user@R1# set mpls interface fxp0.0 disable
user@R1# set ldp interface all
user@R1# set ldp interface fxp0.0 disable
```

7. Specify a time interval for which the targeted LDP sessions are kept up when the remote LFA goes down, and specify a maximum number of automatically, targeted LDP sessions to optimize the use of memory.

```
[edit protocols ldp]
user@R1# set auto-targeted-session
user@R1# set auto-targeted-session teardown-delay 60
user@R1# set auto-targeted-session maximum-sessions 20
```

8. (Optional) Enable forwarding equivalence class (FEC) deaggregation, which results in faster global convergence.

```
[edit protocols ldp]
user@R1# set deaggregate
```

9. To enable Packet Forwarding Engine local repair, establish a policy that forces the routing protocol process to install all the next hops for a given route.

This policy ensures that the backup route is installed in the forwarding table used by the Packet Forwarding Engine to forward traffic to a given destination.

```
[edit policy-options]
user@R1# set policy-options policy-statement ecmp term 1
user@R1# set then load-balance per-packet
```

10. Apply the policy to the forwarding table of the local router with the export statement.

```
[edit routing-options forwarding-table]
user@R1# set export ecmp
```



## Results

From configuration mode, confirm your configuration by entering the **show interfaces**, **show protocols**, **show policy-options**, and **show routing-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces
ge-1/0/0 {
  unit 1 {
    description R1->R2;
    family inet {
      address 10.1.1.1/24;
    }
    family iso;
    family mpls;
  }
}
ge-1/5/0 {
  unit 12 {
    description R1->R6;
    family inet {
      address 10.1.6.12/24;
    }
    family iso;
    family mpls;
  }
}
lo0 {
  unit 10 {
    family inet {
      address 10.255.102.128/32;
    }
    family iso {
      address 49.0001.1720.1600.1030.00;
    }
  }
}
```

```
user@R1# show protocols
mpls {
  interface all;
```

```

    interface fxp0.0 {
        disable;
    }
}
isis {
    spf-options delay 1000;
    backup-spf-options {
        remote-backup-calculation;
        node-link-degradation;
    }
    interface ge-1/0/0.1;
    interface ge-1/5/0.12; {
        link-protection;
    }
    interface all {
        node-link-protection;
        level 2 metric 10;
    }
    interface fxp0.0 {
        disable;
    }
    interface lo0.12 {
        passive;
    }
}
ldp {
    auto-targeted-session {
        teardown-delay 60;
        maximum-sessions 20;
    }
    deaggregate;
    interface all;
    interface fxp0.0 {
        disable;
    }
}

```

```

user@R1# show policy-options
policy-options {
    policy-statement ecmp {
        term 1 {

```

```
        then {  
            load-balance per-packet;  
        }  
    }  
}
```

```
user@R1# show routing-options  
forwarding-table {  
    export ecmp;  
}
```

If you are done configuring the device, enter `commit` from the configuration mode.

## Verification

### IN THIS SECTION

- [Verifying the Routes | 373](#)
- [Verifying the IS-IS Routes | 375](#)
- [Verifying the LDP Routes | 376](#)
- [Verifying the Designated Backup Path Node | 377](#)

Confirm that the configuration is working properly.

### Verifying the Routes

#### Purpose

Verify that the expected routes are learned.

## Action

On Device R1, from operational mode, run the **show route** command to display the routes in the routing table.

```
user@R1> show route 10.1.4/24
```

```
inet.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
10.1.4.0/24          *[IS-IS/15] 11:37:58, metric 30
                    > to 10.1.6.11 via ge-1/5/0
                    to 10.1.1.2 via ge-1/0/0, Push 299824
```

```
user@R1> show route 10.1.4/24 detail
```

```
inet.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)
10.1.4.0/24 (1 entry, 1 announced)
State: <FlashAll>
*IS-IS Preference: 15
  Level: 1
  Next hop type: Router, Next hop index: 262154
  Address: 0x98047cc
  Next-hop reference count: 8
  Next hop: 10.1.6.11 via ge-1/5/0 weight 0x1, selected
  Session Id: 0x14b
  Next hop: 10.1.1.2 via ge-1/0/0 weight 0x101 uflags Remote neighbor path
  Label operation: Push 299824
  Label TTL action: prop-ttl
  Load balance label: Label 299824: None;
  Session Id: 0x142
  State:<Active Int>
  Age: 11:38:00
  Metric: 30
  Validation State: unverified
  Task: IS-IS
  Announcement bits (3): 0-LDP 1-IS-IS 3-KRT
  AS path: I
```

### Meaning

The output shows all the routes in the routing table of Device R1.

### Verifying the IS-IS Routes

### Purpose

Display all the LDP backup routes in the IS-IS routing table of Device R1.

### Action

On Device R1, from operational mode, run the **show isis route** command to display the routes in the IS-IS routing table.

```

user@R1> show isis route
IS-IS routing table          Current version: L1: 558 L2: 564
IPv4/IPv6 Routes
-----
Prefix            L Version  Metric Type Interface    NH   Via              Backup Score
10.1.2.0/24        1      558      20 int  lt-1/2/0.1      IPV4 tp3-R2
10.1.3.0/24        1      558      30 int  lt-1/2/0.1      IPV4 tp3-R2
10.1.4.0/24        1      558      30 int  lt-1/2/0.12     IPV4 tp3-R6
                  1t-1/2/0.1      LSP  LDP->tp3-R4(10.255.102.156)
10.1.5.0/24        1      558      20 int  lt-1/2/0.12     IPV4 tp3-R6
                  1t-1/2/0.1      LSP  LDP->tp3-R4(10.255.102.156)
10.255.102.136/32  1      558      10 int  lt-1/2/0.12     IPV4 tp3-R6
                  1t-1/2/0.1      LSP  LDP->tp3-R4(10.255.102.156)
10.255.102.146/32  1      558      20 int  lt-1/2/0.1      IPV4 tp3-R2
10.255.102.156/32  1      558      30 int  lt-1/2/0.1      IPV4 tp3-R2
                  1t-1/2/0.12     IPV4 tp3-R6
10.255.102.166/32  1      558      20 int  lt-1/2/0.12     IPV4 tp3-R6
                  1t-1/2/0.1      LSP  LDP->tp3-R4(10.255.102.156)
10.255.102.178/32  1      558      10 int  lt-1/2/0.1      IPV4 tp3-R2

```

### Meaning

The output shows all the LDP backup routes in the IS-IS routing table of Device R1.

## Verifying the LDP Routes

### Purpose

Verify the automatically targeted LDP routes.

### Action

From operational mode, enter the **show ldp session auto-targeted detail** command.

```
user@R1> show ldp session auto-targeted detail

Address: 10.255.102.156, State: Operational, Connection: Open, Hold time: 21
  Session ID: 10.255.102.128:0--10.255.102.156:0
  Next keepalive in 1 seconds
  Passive, Maximum PDU: 4096, Hold time: 30, Neighbor count: 1
  Neighbor types: auto-targeted
  Keepalive interval: 10, Connect retry interval: 1
  Local address: 10.255.102.128, Remote address: 10.255.102.156
  Up for 11:38:23
  Capabilities advertised: none
  Capabilities received: none
  Protection: disabled
  Session flags: none
  Local - Restart: disabled, Helper mode: enabled
  Remote - Restart: disabled, Helper mode: enabled
  Local maximum neighbor reconnect time: 120000 msec
  Local maximum neighbor recovery time: 240000 msec
  Local Label Advertisement mode: Downstream unsolicited
  Remote Label Advertisement mode: Downstream unsolicited
  Negotiated Label Advertisement mode: Downstream unsolicited
  MTU discovery: disabled
  Nonstop routing state: Not in sync
  Next-hop addresses received:
    10.1.3.6
    10.1.4.7
    10.255.102.156
```

### Meaning

The output shows automatically targeted LDP next hops.

## Verifying the Designated Backup Path Node

### Purpose

Display the remote LFA next hop determined for a given destination.

### Action

From operational mode, enter the **show isis backup spf results** command.

```
user@R1> show isis backup spf results R6
IS-IS level 1 SPF results:
R6.00
  Primary next-hop: ge-1/5/0, IPV4, R6, SNPA: 0:5:85:88:f0:bc
  Root: R6, Root Metric: 10, Metric: 0, Root Preference: 0x0
    Not eligible, IPV4, Reason: Primary next-hop link fate sharing
  Root: R2, Root Metric: 10, Metric: 20, Root Preference: 0x0
    track-item: R6.00-00
    track-item: R1.00-00
    Not eligible, IPV4, Reason: Path loops
  Root: R4, Root Metric: 30, Metric: 20, Root Preference: 0x0
    track-item: R6.00-00
    track-item: R4.00-00
    Eligible, Backup next-hop: ge-1/0/0, LSP, LDP->R4(10.255.102.156), Prefixes: 2
  1 nodes

IS-IS level 2 SPF results:
R6.00
  Primary next-hop: ge-1/5/0, IPV4, R6, SNPA: 0:5:85:88:f0:bc
  Root: R6, Root Metric: 10, Metric: 0, Root Preference: 0x0
    Not eligible, IPV4, Reason: Primary next-hop link fate sharing
  Root: R2, Root Metric: 10, Metric: 20, Root Preference: 0x0
    track-item: R6.00-00
    track-item: R1.00-00
    Not eligible, IPV4, Reason: Path loops
  Root: R4, Root Metric: 30, Metric: 20, Root Preference: 0x0
    track-item: R6.00-00
    track-item: R4.00-00
    Eligible, Backup next-hop: ge-1/0/0, LSP, LDP->R4(10.255.102.156), Prefixes: 0
  1 nodes
```

## Meaning

The output indicates whether a specific interface or node has been designated as a remote backup path and why.

## RELATED DOCUMENTATION

[Understanding Remote LFA over LDP Tunnels in IS-IS Networks | 357](#)

*auto-targeted-session*

## Understanding Weighted ECMP Traffic Distribution on One-Hop IS-IS Neighbors

Equal-cost multipath (ECMP) is a popular technique to load balance traffic across multiple paths. With ECMP enabled, if paths to a remote destination have the same cost, then traffic is distributed between them in equal proportion. Equal distribution of traffic across multiple paths is not desirable if the local links to adjacent routers towards the ultimate destination have unequal capacity. Typically the traffic distribution between two links is equal and the link utilization is the same. However, if the capacity of an aggregated Ethernet bundle changes, equal traffic distribution results in imbalance of link utilization. In this case, weighted ECMP enables load balancing of traffic between equal cost paths in proportion to the capacity of the local links.

Taking as an example, there are two devices interconnected with an aggregated Ethernet bundle with four links and a single link of the same cost. Under normal conditions, both the AE bundle and the single link is utilized evenly to distribute traffic. However, if a link in the AE bundle goes down, there is a change in the link capacity that results in uneven link utilization. Weighted ECMP load balances traffic between the equal cost paths in proportion to the capacity of the local links. In this case, traffic is distributed in 30/40 proportion between the AE bundle and the single link.



**NOTE:** This feature provides weighted ECMP routing to IS-IS neighbors that are one hop away. Junos OS supports this feature on immediately connected routers only and does not support weighted ECMP on multihop routers, that is, on routers that are more than one hop away.

To enable weighted ECMP traffic distribution on directly connected IS-IS neighbors, configure `weighted one-hop` statement at the `[edit protocols isis spf-options multipath]` hierarchy level. Weighted ECMP is currently supported for the IS-IS protocol only.





**NOTE:** You must configure per-packet load balancing policy before configuring this feature. WECMP will be operational if per-packet load balancing policy is in place,

Weighted ECMP feature also supports IS-IS SPRING based next hop addresses.



**NOTE:** For logical interfaces, you must configure interface bandwidth to distribute traffic across equal cost multipaths based on the underlying physical interface bandwidth. If you do not configure the logical bandwidth for each logical interface, Junos OS assumes that the entire bandwidth of the physical interface is available for each logical interface.

## RELATED DOCUMENTATION

*multipath*

[Example: Weighted ECMP Traffic Distribution on One-Hop IS-IS Neighbors | 379](#)

## Example: Weighted ECMP Traffic Distribution on One-Hop IS-IS Neighbors

### IN THIS SECTION

- [Requirements | 379](#)
- [Overview | 380](#)
- [Configuration | 381](#)
- [Verification | 396](#)

This example shows how to configure weighted equal cost multipath (ECMP) routing for distributing traffic to IS-IS neighbors that are one hop away to ensure optimal load balancing. Weighted ECMP routing distributes traffic unequally over multiple paths for better load balancing. However, weighted ECMP routing is more efficient than equal distribution of traffic during per-packet load balancing.

### Requirements

This example uses the following hardware and software components:

- Two MX Series routers
- Junos OS Release 15.1F4 or later

Before you configure weighted ECMP in an IS-IS network, make sure you :

1. Configure IP addresses on the device interfaces.
2. Configure IS-IS.
3. Configure load balancing
4. Configure a per-packet load balancing policy.

## Overview

### IN THIS SECTION

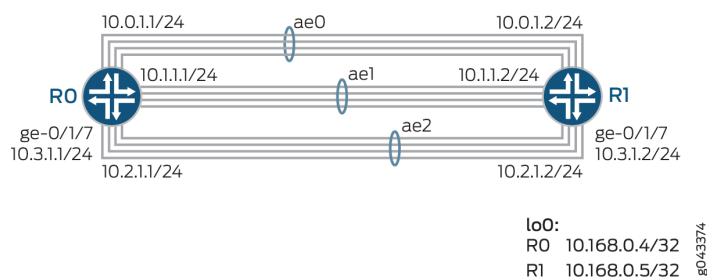
- [Topology | 380](#)

Beginning with Junos OS Release 15.1F4, you can configure the IS-IS protocol to get the logical interface bandwidth information associated with the gateways of equal-cost multipath (ECMP) next hop. During per-packet load balancing, traffic distribution is based on the available bandwidth to facilitate optimal bandwidth usage for incoming traffic on an ECMP path of one hop distance. The Packet Forwarding Engine does not distribute the traffic equally, but considers the balance values and distributes the traffic according to the bandwidth availability. However, this feature is not available for ECMP paths that are more than one hop away.

### Topology

In [Figure 30 on page 381](#), three aggregated Ethernet bundles ae0, ae1, and ae2 with four links each, are configured between Router R0 and Router R1. The Packet Forwarding Engine distributes traffic unequally between the three Ethernet bundles when one of the links goes down, depending on the available bandwidth.

Figure 30: Weighted ECMP Traffic Distribution on One Hop IS-IS Neighbors



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 381](#)
- [Configuring Router R0 | 385](#)
- [Results | 390](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter `commit` from configuration mode.

#### Router R0

```
set interfaces ge-1/1/4 description "LinkID: R0R1-1"
set interfaces ge-0/0/0 description "LinkID: R0R1-2"
set interfaces ge-1/2/1 description "LinkID: R0R1-3"
set interfaces ge-1/2/2 description "LinkID: R0R1-4"
set interfaces ge-1/2/0 description "LinkID: R0R1-5"
set interfaces ge-1/2/3 description "LinkID: R0R1-6"
set interfaces ge-0/1/6 description "LinkID: R0R1-7"
set interfaces ge-1/1/6 description "LinkID: R0R1-8"
set interfaces ge-1/1/5 description "LinkID: R0R1-9"
set interfaces ge-1/3/3 description "LinkID: R0R1-10"
set interfaces ge-1/2/8 description "LinkID: R0R1-11"
set interfaces ge-0/1/8 description "LinkID: R0R1-12"
set interfaces ge-0/1/7 description "LinkID: R0R1-13"
```

```

set interfaces ge-0/0/1 description "LinkID: R0RT0"
set chassis maximum-ecmp 64
set chassis redundancy graceful-switchover
set chassis aggregated-devices ethernet device-count 64
set interfaces ge-0/0/1 unit 0 family inet address 21.1.1.1/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-1/1/4 gigether-options 802.3ad ae0
set interfaces ge-0/0/0 gigether-options 802.3ad ae0
set interfaces ge-1/2/1 gigether-options 802.3ad ae0
set interfaces ge-1/2/2 gigether-options 802.3ad ae0
set interfaces ge-1/2/0 gigether-options 802.3ad ae1
set interfaces ge-1/2/3 gigether-options 802.3ad ae1
set interfaces ge-0/1/6 gigether-options 802.3ad ae1
set interfaces ge-1/1/6 gigether-options 802.3ad ae1
set interfaces ge-1/1/5 gigether-options 802.3ad ae2
set interfaces ge-1/3/3 gigether-options 802.3ad ae2
set interfaces ge-1/2/8 gigether-options 802.3ad ae2
set interfaces ge-0/1/8 gigether-options 802.3ad ae2
set interfaces ge-0/1/7 unit 0 family inet address 10.3.1.1/24
set interfaces ge-0/1/7 unit 0 family iso
set interfaces ae0 aggregated-ether-options minimum-links 1
set interfaces ae0 aggregated-ether-options lacp active
set interfaces ae0 unit 0 family inet address 10.0.1.1/24
set interfaces ae0 unit 0 family iso
set interfaces ae1 vlan-tagging
set interfaces ae1 aggregated-ether-options minimum-links 1
set interfaces ae1 aggregated-ether-options lacp active
set interfaces ae1 unit 0 family inet address 10.1.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 1;
set interfaces ae1 unit 1 family inet address 13.1.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 2;
set interfaces ae1 unit 1 family inet address 13.2.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 3;
set interfaces ae1 unit 1 family inet address 13.3.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 4;

```

```

set interfaces ae1 unit 1 family inet address 13.4.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 5;
set interfaces ae1 unit 1 family inet address 13.5.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 6;
set interfaces ae1 unit 1 family inet address 13.6.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 7;
set interfaces ae1 unit 1 family inet address 13.7.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 8;
set interfaces ae1 unit 1 family inet address 13.8.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 9;
set interfaces ae1 unit 1 family inet address 13.9.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae1 unit 1 bandwidth 1g;
set interfaces ae1 unit 1 vlan-id 10;
set interfaces ae1 unit 1 family inet address 13.10.1.1/24
set interfaces ae1 unit 0 family iso
set interfaces ae2 aggregated-ether-options minimum-links 2
set interfaces ae2 aggregated-ether-options lacp active
set interfaces ae2 unit 0 family inet address 10.2.1.1/24
set interfaces ae2 unit 0 family iso
set interfaces lo0 unit 0 family iso address 49.0001.0102.5516.3127.00
set routing-options forwarding-table export pplb
set protocols isis interface ge-0/0/1.0
set protocols isis interface ge-0/1/7.0 level 1 metric 20
set protocols isis interface ge-0/1/7.0 level 2 metric 20
set protocols isis interface ae0.0 node-link-protection
set protocols isis interface ae1.0
set protocols isis interface ae2.0
set protocols isis interface lo0.0
set policy-options policy-statement pplb then load-balance per-packet
set protocols isis spf-options multipath weighted one-hop

```

## Router R1

```

set interfaces ge-1/1/4 description "LinkID: R0R1-1"
set interfaces ge-0/0/0 description "LinkID: R0R1-2"
set interfaces ge-1/2/1 description "LinkID: R0R1-3"
set interfaces ge-1/2/2 description "LinkID: R0R1-4"
set interfaces ge-1/2/0 description "LinkID: R0R1-5"
set interfaces ge-1/2/3 description "LinkID: R0R1-6"
set interfaces ge-0/1/6 description "LinkID: R0R1-7"
set interfaces ge-1/1/6 description "LinkID: R0R1-8"
set interfaces ge-1/1/5 description "LinkID: R0R1-9"
set interfaces ge-1/3/3 description "LinkID: R0R1-10"
set interfaces ge-1/2/8 description "LinkID: R0R1-11"
set interfaces ge-0/1/8 description "LinkID: R0R1-12"
set interfaces ge-0/1/7 description "LinkID: R0R1-13"
set interfaces ge-0/1/0 description "LinkID: R1RT0"
set chassis aggregated-devices ethernet device-count 64
set interfaces ge-0/1/0 unit 0 family inet address 22.1.1.1/24
set interfaces ge-0/1/0 unit 0 family iso
set interfaces ge-1/1/4 gigether-options 802.3ad ae0
set interfaces ge-0/0/0 gigether-options 802.3ad ae0
set interfaces ge-1/2/1 gigether-options 802.3ad ae0
set interfaces ge-1/2/2 gigether-options 802.3ad ae0
set interfaces ge-1/2/0 gigether-options 802.3ad ae1
set interfaces ge-1/2/3 gigether-options 802.3ad ae1
set interfaces ge-0/1/6 gigether-options 802.3ad ae1
set interfaces ge-1/1/6 gigether-options 802.3ad ae1
set interfaces ge-1/1/5 gigether-options 802.3ad ae2
set interfaces ge-1/3/3 gigether-options 802.3ad ae2
set interfaces ge-1/2/8 gigether-options 802.3ad ae2
set interfaces ge-0/1/8 gigether-options 802.3ad ae2
set interfaces ge-0/1/7 unit 0 family inet address 10.3.1.2/24
set interfaces ge-0/1/7 unit 0 family iso
set interfaces ae0 aggregated-ether-options minimum-links 1
set interfaces ae0 aggregated-ether-options lacp active
set interfaces ae0 unit 0 family inet address 10.0.1.2/24
set interfaces ae0 unit 0 family iso
set interfaces ae1 aggregated-ether-options minimum-links 1
set interfaces ae1 aggregated-ether-options lacp active
set interfaces ae1 unit 0 family inet address 10.1.1.2/24
set interfaces ae1 unit 0 family iso
set interfaces ae2 aggregated-ether-options minimum-links 2

```

```

set interfaces ae2 aggregated-ether-options lacp active
set interfaces ae2 unit 0 family inet address 10.2.1.2/24
set interfaces ae2 unit 0 family iso
set interfaces lo0 unit 0 family iso address 49.0001.0102.5516.3130.00
set protocols isis export from-static
set protocols isis interface ge-0/1/0.0
set protocols isis interface ge-0/1/7.0
set protocols isis interface ae0.0
set protocols isis interface ae1.0
set protocols isis interface ae2.0
set protocols isis interface lo0.0
set policy-options policy-statement from-static from protocol static
set policy-options policy-statement from-static then accept

```

## Configuring Router R0

### Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure Router R0:



**NOTE:** Repeat this procedure for Router R1 after modifying the appropriate interface names, addresses, and other parameters.

1. Specify the maximum number of weighted ECMP interfaces that you want to configure. Enable graceful switchover and specify the number of aggregated Ethernet interfaces to be created.

```

[edit chassis]
user@R0# set maximum-ecmp 64
user@R0# set redundancy graceful-switchover
user@R0# set aggregated-devices ethernet device-count 64

```

2. Configure the interfaces with multiple links to the same destination for load balancing traffic.

```

[edit interfaces]
user@R0# set ge-1/1/4 description "LinkID: R0R1-1"
user@R0# set ge-0/0/0 description "LinkID: R0R1-2"

```

```

user@R0# set ge-1/2/1 description "LinkID: R0R1-3"
user@R0# set ge-1/2/2 description "LinkID: R0R1-4"
user@R0# set ge-1/2/0 description "LinkID: R0R1-5"
user@R0# set ge-1/2/3 description "LinkID: R0R1-6"
user@R0# set ge-0/1/6 description "LinkID: R0R1-7"
user@R0# set ge-1/1/6 description "LinkID: R0R1-8"
user@R0# set ge-1/1/5 description "LinkID: R0R1-9"
user@R0# set ge-1/3/3 description "LinkID: R0R1-10"
user@R0# set ge-1/2/8 description "LinkID: R0R1-11"
user@R0# set ge-0/1/8 description "LinkID: R0R1-12"
user@R0# set ge-0/1/7 description "LinkID: R0R1-13"
user@R0# set ge-0/0/1 description "LinkID: R0RT0"

```

3. Configure logical interfaces with appropriate bandwidth based on the underlying physical bandwidth.

```

user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 1;
user@R0# set ae1 unit 1 family inet address 13.1.1.1/24
user@R0# set ae1 unit 0 family iso
user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 2;
user@R0# set ae1 unit 1 family inet address 13.2.1.1/24
user@R0# set ae1 unit 0 family iso
user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 3;
user@R0# set ae1 unit 1 family inet address 13.3.1.1/24
user@R0# set ae1 unit 0 family iso
user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 4;
user@R0# set ae1 unit 1 family inet address 13.4.1.1/24
user@R0# set ae1 unit 0 family iso
user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 5;
user@R0# set ae1 unit 1 family inet address 13.5.1.1/24
user@R0# set ae1 unit 0 family iso
user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 6;
user@R0# set ae1 unit 1 family inet address 13.6.1.1/24
user@R0# set ae1 unit 0 family iso
user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 7;

```



```

user@R0# set ae1 unit 1 family inet address 13.7.1.1/24
user@R0# set ae1 unit 0 family iso
user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 8;
user@R0# set ae1 unit 1 family inet address 13.8.1.1/24
user@R0# set ae1 unit 0 family iso
user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 9;
user@R0# set ae1 unit 1 family inet address 13.9.1.1/24
user@R0# set ae1 unit 0 family iso
user@R0# set ae1 unit 1 bandwidth 1g;
user@R0# set ae1 unit 1 vlan-id 10;
user@R0# set ae1 unit 1 family inet address 13.10.1.1/24
user@R0# set ae1 unit 0 family iso

```



**NOTE:** For logical interfaces, configure interface bandwidth to distribute traffic across equal-cost multipaths based on the underlying operational interface bandwidth. When you configure multiple logical interfaces on a single interface, configure appropriate logical bandwidth for each logical interface to see the desired traffic distribution over the logical interfaces.

4. Configure IP addresses on the interfaces with either IPv4 or IPv6 addresses, as per your network requirements.

```

[edit interfaces]
user@R0# set ge-0/0/1 unit 0 family inet address 21.1.1.1/24
user@R0# set ge-0/0/1 unit 0 family iso
user@R0# set ge-0/1/7 unit 0 family inet address 10.3.1.1/24
user@R0# set ge-0/1/7 unit 0 family iso

```

5. Configure the four member links of the ae0 aggregated Ethernet bundle.

```

[edit interfaces]
user@R0# set ge-1/1/4 gigether-options 802.3ad ae0
user@R0# set ge-0/0/0 gigether-options 802.3ad ae0
user@R0# set ge-1/2/1 gigether-options 802.3ad ae0
user@R0# set ge-1/2/2 gigether-options 802.3ad ae0

```

6. Configure the four member links of the ae1 aggregated Ethernet bundle.

```
[edit interfaces]
user@R0# set ge-1/2/0 gigether-options 802.3ad ae1
user@R0# set ge-1/2/3 gigether-options 802.3ad ae1
user@R0# set ge-0/1/6 gigether-options 802.3ad ae1
user@R0# set ge-1/1/6 gigether-options 802.3ad ae1
```

7. Configure the four member links of the ae2 aggregated Ethernet bundle.

```
[edit interfaces]
user@R0# set ge-1/1/5 gigether-options 802.3ad ae2
user@R0# set ge-1/3/3 gigether-options 802.3ad ae2
user@R0# set ge-1/2/8 gigether-options 802.3ad ae2
user@R0# set ge-0/1/8 gigether-options 802.3ad ae2
```

8. Configure IP address and the Link Aggregation Control Protocol (LACP) for ae0 aggregated Ethernet interface.

```
[edit interfaces]
user@R0# set ae0 aggregated-ether-options minimum-links 1
user@R0# set ae0 aggregated-ether-options lacp active
user@R0# set ae0 unit 0 family inet address 10.0.1.1/24
user@R0# set ae0 unit 0 family iso
```

9. Configure IP address and the Link Aggregation Control Protocol (LACP) for ae1 aggregated Ethernet interface.

```
[edit interfaces]
user@R0# set ae1 aggregated-ether-options minimum-links 1
user@R0# set ae1 aggregated-ether-options lacp active
user@R0# set ae1 unit 0 family inet address 10.1.1.1/24
user@R0# set ae1 unit 0 family iso
```

10. Configure IP address and the Link Aggregation Control Protocol (LACP) for ae2 aggregated Ethernet interface.

```
[edit interfaces]
user@R0# set ae2 aggregated-ether-options minimum-links 2
user@R0# set ae2 aggregated-ether-options lacp active
user@R0# set ae2 unit 0 family inet address 10.2.1.1/24
user@R0# set ae2 unit 0 family iso
```

11. Configure the loopback interface address and iso family address.

```
[edit interfaces]
user@R0# set lo0 unit 0 family inet address 10.168.0.4/32
user@R0# set lo0 unit 0 family iso address 49.0001.0102.5516.3127.00
```

12. Configure IS-IS on all the interfaces and on the AE bundles.

```
[edit protocols]
user@R0# set isis interface ge-0/0/1.0
user@R0# set isis interface ge-0/1/7.0 level 1 metric 20
user@R0# set isis interface ge-0/1/7.0 level 2 metric 20
user@R0# set isis interface ae0.0 node-link-protection
user@R0# set isis interface ae1.0
user@R0# set isis interface ae2.0
user@R0# set isis interface lo0.0
```

13. Configure per-packet load balancing.

```
[edit policy-options]
user@R0# set policy-statement pplb then load-balance per-packet
```

14. Apply per-packet load balancing policy.

```
[edit routing-options]
user@R0# set forwarding-table
user@R0# set export pplb
```

## 15. Enable weighted ECMP traffic distribution on directly connected IS-IS neighbors.

```
[edit protocols isis]
user@R0# set spf-options multipath weighted one-hop
```

### Results

From configuration mode, confirm your configuration by entering the `show chassis`, `show interfaces`, `show protocols`, `show policy-options`, and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@R0# show chassis
maximum-ecmp 64;
redundancy graceful-switchover;
aggregated-devices ethernet device-count 64;
```

```
[edit]
user@R0# show interfaces
ge-0/0/0 {
    description "LinkID: R0R1-2";
    gigether-options {
        802.3ad ae0;
    }
}
ge-0/0/1 {
    description "LinkID: R0RT0";
    unit 0 {
        family inet {
            address 21.1.1.1/24;
        }
        family iso;
    }
}
ge-0/1/6 {
    description "LinkID: R0R1-7";
    gigether-options {
        802.3ad ae1;
    }
}
```

```

}
ge-0/1/7 {
    description "LinkID: R0R1-13";
    unit 0 {
        family inet {
            address 10.3.1.1/24;
        }
        family iso;
    }
}
ge-0/1/8 {
    description "LinkID: R0R1-12";
    gigether-options {
        802.3ad ae2;
    }
}
ge-1/1/4 {
    description "LinkID: R0R1-1";
    gigether-options {
        802.3ad ae0;
    }
}
ge-1/1/5 {
    description "LinkID: R0R1-9";
    gigether-options {
        802.3ad ae2;
    }
}
ge-1/1/6 {
    description "LinkID: R0R1-8";
    gigether-options {
        802.3ad ae1;
    }
}
ge-1/2/0 {
    description "LinkID: R0R1-5";
    gigether-options {
        802.3ad ae1;
    }
}
ge-1/2/1 {
    description "LinkID: R0R1-3";
    gigether-options {

```

```

        802.3ad ae0;
    }
}
ge-1/2/2 {
    description "LinkID: R0R1-4";
    gigether-options {
        802.3ad ae0;
    }
}
ge-1/2/3 {
    description "LinkID: R0R1-6";
    gigether-options {
        802.3ad ae1;
    }
}
ge-1/2/8 {
    description "LinkID: R0R1-11";
    gigether-options {
        802.3ad ae2;
    }
}
ge-1/3/3 {
    description "LinkID: R0R1-10";
    gigether-options {
        802.3ad ae2;
    }
}
ae0 {
    aggregated-ether-options {
        minimum-links 1;
        lacp {
            active;
        }
    }
    unit 0 {
        family inet {
            address 10.0.1.1/24;
        }
        family iso;
    }
}
ae1 {
    vlan-tagging;

```

```
aggregated-ether-options {  
    minimum-links 3;  
    lacp {  
        active;  
    }  
}  
unit 0 {  
    family inet {  
        address 10.1.1.1/24;  
    }  
    family iso;  
}  
unit 1 {  
    bandwidth 1g;  
    vlan-id 1;  
    family inet {  
        address 13.1.1.1/24;  
    }  
    family iso;  
}  
unit 2 {  
    bandwidth 1g;  
    vlan-id 2;  
    family inet {  
        address 13.2.1.1/24;  
    }  
    family iso;  
}  
unit 3 {  
    bandwidth 1g;  
    vlan-id 3;  
    family inet {  
        address 13.3.1.1/24;  
    }  
    family iso;  
}  
unit 4 {  
    bandwidth 1g;  
    vlan-id 4;  
    family inet {  
        address 13.4.1.1/24;  
    }  
    family iso;  
}
```

```
}  
unit 5 {  
    bandwidth 1g;  
    vlan-id 5;  
    family inet {  
        address 13.5.1.1/24;  
    }  
    family iso;  
}  
unit 6 {  
    bandwidth 1g;  
    vlan-id 6;  
    family inet {  
        address 13.6.1.1/24;  
    }  
    family iso;  
}  
unit 7 {  
    bandwidth 1g;  
    vlan-id 7;  
    family inet {  
        address 13.7.1.1/24;  
    }  
    family iso;  
}  
unit 8 {  
    bandwidth 1g;  
    vlan-id 8;  
    family inet {  
        address 13.8.1.1/24;  
    }  
    family iso;  
}  
unit 9 {  
    bandwidth 1g;  
    vlan-id 9;  
    family inet {  
        address 13.9.1.1/24;  
    }  
    family iso;  
}  
unit 10 {  
    bandwidth 1g;
```



```

        vlan-id 10;
        family inet {
            address 13.10.1.1/24;
        }
        family iso;
    }
}
ae2 {
    aggregated-ether-options {
        minimum-links 2;
        lacp {
            active;
        }
    }
    unit 0 {
        family inet {
            address 10.2.1.1/24;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.0.4/32;
        }
        family iso {
            address 49.0001.0102.5516.3127.00;
        }
    }
}
}

```

```

[edit]
user@R0# show protocols
isis {
    spf-options {
        multi-path {
            weighted {
                one-hop;
            }
        }
    }
}

```

```

    }
}

```

```

[edit]
user@R01# show policy-options
policy-statement pplb {
    then {
        load-balance per-packet;
    }
}

```

```

[edit]
user@R0# show routing-options
forwarding-table {
    export pplb;
}

```

## Verification

### IN THIS SECTION

- [Verifying Equal Distribution of Traffic Over Equal-Cost Multiple Paths | 396](#)
- [Verifying Unequal Traffic Distribution Over Available Bandwidth | 401](#)
- [Verifying Unequal Traffic Distribution on Logical Interfaces | 406](#)

Confirm that the configuration is working properly.

### Verifying Equal Distribution of Traffic Over Equal-Cost Multiple Paths

#### Purpose

To verify that traffic is equally distributed over the aggregated Ethernet bundles.

## Action

From operational mode, enter the `show route 198.0.0.1 extensive` command.

```

user@R0> show route 198.0.0.1 extensive
inet.0: 10028 destinations, 10029 routes (10027 active, 0 holddown, 1 hidden)
198.0.0.1/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 198.0.0.1/32 -> {10.0.1.2, 10.1.1.2, 10.2.1.2, 10.3.1.2}
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Router, Next hop index: 1048574
        Address: 0x9ec5e10
        Next-hop reference count: 20005
        Next hop: 10.0.1.2 via ae0.0 weight 0x1 balance 33%
        Session Id: 0x1b2
        Next hop: 10.1.1.2 via ae1.0 weight 0x1 balance 33%, selected
        Session Id: 0x1b1
        Next hop: 10.2.1.2 via ae2.0 weight 0x1 balance 33%
        Session Id: 0x1b3
        Next hop: 10.3.1.2 via ge-0/1/7.0 weight 0xf000
        Session Id: 0x1b0
        State: <Active Int>
            Age: 35      Metric: 20
            Validation State: unverified
            Task: IS-IS
            Announcement bits (1): 0-KRT
            AS path: I
user@R0> show interfaces ae0.0 extensive
Logical interface ae0.0 (Index 335) (SNMP ifIndex 625) (Generation 825)
  Flags: Up SNMP-Traps 0x4004000 Encapsulation: ENET2
  Statistics          Packets          pps          Bytes          bps
  Bundle:
    Input :           702             4         207265         4320
    Output:        870567        33801       95801535       29746416
  Adaptive Statistics:
    Adaptive Adjusts:           0
    Adaptive Scans  :           0
    Adaptive Updates:           0
  Link:
    ge-0/0/0.0
    Input :           149             1         17924          992

```

```

Output:      218927      8586      24081728      7556344
ge-1/1/4.0
Input :       134        1        16616         992
Output:     201384      7781      22152240      6847320
ge-1/2/1.0
Input :       136        1        16864         992
Output:     212760      8238      23443069      7250056
ge-1/2/2.0
Input :       283        1       155861        1344
Output:     237496      9196      26124498      8092696

```

Aggregate member links: 4

LACP info:	Role	System priority	System identifier	Port priority	Port number	Port key
ge-0/0/0.0	Actor	127	3c:61:04:2f:c7:c0	127	1	1
ge-0/0/0.0	Partner	127	3c:61:04:2d:9f:c0	127	1	1
ge-1/1/4.0	Actor	127	3c:61:04:2f:c7:c0	127	4	1
ge-1/1/4.0	Partner	127	3c:61:04:2d:9f:c0	127	4	1
ge-1/2/1.0	Actor	127	3c:61:04:2f:c7:c0	127	8	1
ge-1/2/1.0	Partner	127	3c:61:04:2d:9f:c0	127	8	1
ge-1/2/2.0	Actor	127	3c:61:04:2f:c7:c0	127	9	1
ge-1/2/2.0	Partner	127	3c:61:04:2d:9f:c0	127	9	1

LACP Statistics:	LACP Rx	LACP Tx	Unknown Rx	Illegal Rx
ge-0/0/0.0	130	125	0	0
ge-1/1/4.0	127	121	0	0
ge-1/2/1.0	127	123	0	0
ge-1/2/2.0	128	123	0	0

Marker Statistics:	Marker Rx	Resp Tx	Unknown Rx	Illegal Rx
ge-0/0/0.0	0	0	0	0
ge-1/1/4.0	0	0	0	0
ge-1/2/1.0	0	0	0	0
ge-1/2/2.0	0	0	0	0

Protocol inet, MTU: 1500, Generation: 1699, Route table: 0

Flags: Sendbcst-pkt-to-re

Addresses, Flags: Is-Preferred Is-Primary

Destination: 10.0.1/24, Local: 10.0.1.1, Broadcast: 10.0.1.255, Generation: 1501

Protocol iso, MTU: 1497, Generation: 1700, Route table: 0

Flags: Is-Primary

Protocol multiservice, MTU: Unlimited, Generation: 1701, Route table: 0

Flags: Is-Primary

Policer: Input: \_\_default\_arp\_policer\_\_

```
user@R0> show interfaces ae1.0 extensive
```

```
Logical interface ae1.0 (Index 336) (SNMP ifIndex 666) (Generation 826)
```

```
Flags: Up SNMP-Traps 0x4004000 Encapsulation: ENET2
```

```
Statistics      Packets      pps      Bytes      bps
```

```
Bundle:
```

```
Input :          707          4      206275      3968
```

```
Output:      849981      32979      93602009      29023264
```

```
Adaptive Statistics:
```

```
Adaptive Adjusts:      0
```

```
Adaptive Scans :      0
```

```
Adaptive Updates:      0
```

```
Link:
```

```
ge-0/1/6.0
```

```
Input :          148          1      17800      992
```

```
Output:      198301      7819      21812806      6880984
```

```
ge-1/1/6.0
```

```
Input :          134          1      16616      992
```

```
Output:      209149      8088      23006390      7117728
```

```
ge-1/2/0.0
```

```
Input :          136          1      16864      992
```

```
Output:      215518      8291      23811445      7296528
```

```
ge-1/2/3.0
```

```
Input :          289          1      154995      992
```

```
Output:      227013      8781      24971368      7728024
```

```
Aggregate member links: 4
```

LACP info:	Role	System priority	System identifier	Port priority	Port number	Port key
ge-0/1/6.0	Actor	127	3c:61:04:2f:c7:c0	127	2	2
ge-0/1/6.0	Partner	127	3c:61:04:2d:9f:c0	127	2	2
ge-1/1/6.0	Actor	127	3c:61:04:2f:c7:c0	127	6	2
ge-1/1/6.0	Partner	127	3c:61:04:2d:9f:c0	127	6	2
ge-1/2/0.0	Actor	127	3c:61:04:2f:c7:c0	127	7	2
ge-1/2/0.0	Partner	127	3c:61:04:2d:9f:c0	127	7	2
ge-1/2/3.0	Actor	127	3c:61:04:2f:c7:c0	127	10	2
ge-1/2/3.0	Partner	127	3c:61:04:2d:9f:c0	127	10	2

LACP Statistics:	LACP Rx	LACP Tx	Unknown Rx	Illegal Rx
ge-0/1/6.0	129	123	0	0
ge-1/1/6.0	127	121	0	0
ge-1/2/0.0	127	123	0	0

```
ge-1/2/3.0          128          123          0          0
Marker Statistics:  Marker Rx      Resp Tx      Unknown Rx   Illegal Rx
ge-0/1/6.0          0           0           0           0
ge-1/1/6.0          0           0           0           0
ge-1/2/0.0          0           0           0           0
ge-1/2/3.0          0           0           0           0
Protocol inet, MTU: 1500, Generation: 1702, Route table: 0
  Flags: Sendbcast-pkt-to-re
  Addresses, Flags: Is-Preferred Is-Primary
    Destination: 10.1.1/24, Local: 10.1.1.1, Broadcast: 10.1.1.255, Generation: 1503
Protocol iso, MTU: 1497, Generation: 1703, Route table: 0
Protocol multiservice, MTU: Unlimited, Generation: 1704, Route table: 0
  Policer: Input: __default_arp_policer__
```

user@R0> **show interfaces ae2.0 extensive**

Logical interface ae2.0 (Index 337) (SNMP ifIndex 961) (Generation 827)

Flags: Up SNMP-Traps 0x4004000 Encapsulation: ENET2

Statistics	Packets	pps	Bytes	bps
Bundle:				
Input :	702	4	224128	3968
Output:	855472	33229	94215862	29243664

Adaptive Statistics:

Adaptive Adjusts:	0
Adaptive Scans :	0
Adaptive Updates:	0

Link:

ge-0/1/8.0				
Input :	137	1	16988	992
Output:	213214	8377	23453540	7372232
ge-1/1/5.0				
Input :	137	1	16988	992
Output:	212174	8244	23339050	7255368
ge-1/2/8.0				
Input :	135	1	16740	992
Output:	210583	8144	23164099	7167296
ge-1/3/3.0				
Input :	293	1	173412	992
Output:	219501	8464	24259173	7448768

Aggregate member links: 4

LACP info:	Role	System	System	Port	Port	Port
------------	------	--------	--------	------	------	------

```

                priority      identifier  priority  number  key
ge-0/1/8.0      Actor        127      3c:61:04:2f:c7:c0      127      3      3
ge-0/1/8.0      Partner      127      3c:61:04:2d:9f:c0      127      3      3
ge-1/1/5.0      Actor        127      3c:61:04:2f:c7:c0      127      5      3
ge-1/1/5.0      Partner      127      3c:61:04:2d:9f:c0      127      5      3
ge-1/2/8.0      Actor        127      3c:61:04:2f:c7:c0      127     11      3
ge-1/2/8.0      Partner      127      3c:61:04:2d:9f:c0      127     11      3
ge-1/3/3.0      Actor        127      3c:61:04:2f:c7:c0      127     12      3
ge-1/3/3.0      Partner      127      3c:61:04:2d:9f:c0      127     12      3
LACP Statistics:  LACP Rx    LACP Tx    Unknown Rx  Illegal Rx
ge-0/1/8.0        127        123         0           0
ge-1/1/5.0       130        123         0           0
ge-1/2/8.0       129        124         0           0
ge-1/3/3.0       129        124         0           0
Marker Statistics: Marker Rx    Resp Tx    Unknown Rx  Illegal Rx
ge-0/1/8.0         0          0          0           0
ge-1/1/5.0         0          0          0           0
ge-1/2/8.0         0          0          0           0
ge-1/3/3.0         0          0          0           0
Protocol inet, MTU: 1500, Generation: 1705, Route table: 0
  Flags: Sendbcst-pkt-to-re
  Addresses, Flags: Is-Preferred Is-Primary
    Destination: 10.2.1/24, Local: 10.2.1.1, Broadcast: 10.2.1.255, Generation: 1505
Protocol iso, MTU: 1497, Generation: 1706, Route table: 0
Protocol multiservice, MTU: Unlimited, Generation: 1707, Route table: 0
  Policer: Input: __default_arp_policer__

```

## Meaning

IS-IS distributes traffic equally when the three aggregated Ethernet bundles have the same bandwidth available.

## Verifying Unequal Traffic Distribution Over Available Bandwidth

### Purpose

To verify that IS-IS distributes traffic unevenly when one of the aggregated link is down during per-packet load balancing depending on the available bandwidth.

## Action

Disable one of the links on the ae0 bundle so that the available bandwidth is 3g on ae0 and 4g on ae1 and ae2. From operational mode, enter the `show route 198.0.0.1 extensive` command.

```

user@R0> show route 198.0.0.1 extensive
inet.0: 10028 destinations, 10029 routes (10027 active, 0 holddown, 1 hidden)
198.0.0.1/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 198.0.0.1/32 -> {10.0.1.2, 10.1.1.2, 10.2.1.2, 10.3.1.2}
  *IS-IS Preference: 18
    Level: 2
    Next hop type: Router, Next hop index: 1048575
    Address: 0x9ec55d0
    Next-hop reference count: 20005
    Next hop: 10.0.1.2 via ae0.0 weight 0x1 balance 27%
    Session Id: 0x1b2
    Next hop: 10.1.1.2 via ae1.0 weight 0x1 balance 36%, selected
    Session Id: 0x1b1
    Next hop: 10.2.1.2 via ae2.0 weight 0x1 balance 36%
    Session Id: 0x1b3
    Next hop: 10.3.1.2 via ge-0/1/7.0 weight 0xf000
    Session Id: 0x1b0
    State:<Active Int>
    Age: 22      Metric: 20
    Validation State: unverified
    Task: IS-IS
    Announcement bits (1): 0-KRT
    AS path: I
user@R0> show interfaces ae0.0 extensive
Logical interface ae0.0 (Index 335) (SNMP ifIndex 625) (Generation 825)
Flags: Up SNMP-Traps 0x4004000 Encapsulation: ENET2
Statistics          Packets          pps          Bytes          bps
Bundle:
  Input :           793             3         218290         2976
  Output:        1617811        27223        178003101       23957320
Adaptive Statistics:
  Adaptive Adjusts:           0
  Adaptive Scans :           0
  Adaptive Updates:           0
Link:
  ge-0/0/0.0

```



```

Input :      182      1      21794      992
Output:    461045    9423    50717650    8292776
ge-1/1/4.0 <-- down
Input :      139      0      17236      0
Output:    241334      0    26546740      0
ge-1/2/1.0
Input :      162      1      20088      992
Output:    444340    8979    48918653    7901976
ge-1/2/2.0
Input :      310      1      159172      992
Output:    471092    8821    51820058    7762568

```

Aggregate member links: 4

LACP info:	Role	System priority	System identifier	Port priority	Port number	Port key
ge-0/0/0.0	Actor	127	3c:61:04:2f:c7:c0	127	1	1
ge-0/0/0.0	Partner	127	3c:61:04:2d:9f:c0	127	1	1
ge-1/1/4.0	Actor	127	3c:61:04:2f:c7:c0	127	4	1
ge-1/1/4.0	Partner	1	00:00:00:00:00:00	1	4	1
ge-1/2/1.0	Actor	127	3c:61:04:2f:c7:c0	127	8	1
ge-1/2/1.0	Partner	127	3c:61:04:2d:9f:c0	127	8	1
ge-1/2/2.0	Actor	127	3c:61:04:2f:c7:c0	127	9	1
ge-1/2/2.0	Partner	127	3c:61:04:2d:9f:c0	127	9	1

LACP Statistics:	LACP Rx	LACP Tx	Unknown Rx	Illegal Rx
ge-0/0/0.0	161	156	0	0
ge-1/1/4.0	151	145	0	0
ge-1/2/1.0	158	154	0	0
ge-1/2/2.0	159	154	0	0

Marker Statistics:	Marker Rx	Resp Tx	Unknown Rx	Illegal Rx
ge-0/0/0.0	0	0	0	0
ge-1/1/4.0	0	0	0	0
ge-1/2/1.0	0	0	0	0
ge-1/2/2.0	0	0	0	0

Protocol inet, MTU: 1500, Generation: 1699, Route table: 0

Flags: Sendbcst-pkt-to-re

Addresses, Flags: Is-Preferred Is-Primary

Destination: 10.0.1/24, Local: 10.0.1.1, Broadcast: 10.0.1.255, Generation: 1501

Protocol iso, MTU: 1497, Generation: 1700, Route table: 0

Flags: Is-Primary

Protocol multiservice, MTU: Unlimited, Generation: 1701, Route table: 0

Flags: Is-Primary

Policer: Input: \_\_default\_arp\_policer\_\_

user@R0> **show interfaces ae1.0 extensive**

Logical interface ae1.0 (Index 336) (SNMP ifIndex 666) (Generation 826)

Flags: Up SNMP-Traps 0x4004000 Encapsulation: ENET2

Statistics	Packets	pps	Bytes	bps
------------	---------	-----	-------	-----

Bundle:

Input :	817	5	219555	4672
---------	-----	---	--------	------

Output:	1756031	<b>35775</b>	193270683	31483104
---------	---------	--------------	-----------	----------

Adaptive Statistics:

Adaptive Adjusts:	0
-------------------	---

Adaptive Scans :	0
------------------	---

Adaptive Updates:	0
-------------------	---

Link:

ge-0/1/6.0

Input :	174	1	21024	992
---------	-----	---	-------	-----

Output:	411469	8414	45261286	7404544
---------	--------	------	----------	---------

ge-1/1/6.0

Input :	159	1	19716	992
---------	-----	---	-------	-----

Output:	433700	8893	47707000	7826296
---------	--------	------	----------	---------

ge-1/2/0.0

Input :	161	1	19964	992
---------	-----	---	-------	-----

Output:	447338	9190	49314819	8087408
---------	--------	------	----------	---------

ge-1/2/3.0

Input :	323	2	158851	1696
---------	-----	---	--------	------

Output:	463524	9278	50987578	8164856
---------	--------	------	----------	---------

Aggregate member links: 4

LACP info:	Role	System priority	System identifier	Port priority	Port number	Port key
ge-0/1/6.0	Actor	127	3c:61:04:2f:c7:c0	127	2	2
ge-0/1/6.0	Partner	127	3c:61:04:2d:9f:c0	127	2	2
ge-1/1/6.0	Actor	127	3c:61:04:2f:c7:c0	127	6	2
ge-1/1/6.0	Partner	127	3c:61:04:2d:9f:c0	127	6	2
ge-1/2/0.0	Actor	127	3c:61:04:2f:c7:c0	127	7	2
ge-1/2/0.0	Partner	127	3c:61:04:2d:9f:c0	127	7	2
ge-1/2/3.0	Actor	127	3c:61:04:2f:c7:c0	127	10	2
ge-1/2/3.0	Partner	127	3c:61:04:2d:9f:c0	127	10	2

LACP Statistics:	LACP Rx	LACP Tx	Unknown Rx	Illegal Rx
------------------	---------	---------	------------	------------

ge-0/1/6.0	159	153	0	0
------------	-----	-----	---	---

```

ge-1/1/6.0          157          151          0          0
ge-1/2/0.0          157          153          0          0
ge-1/2/3.0          158          153          0          0
Marker Statistics:  Marker Rx      Resp Tx      Unknown Rx      Illegal Rx
ge-0/1/6.0          0            0            0            0
ge-1/1/6.0          0            0            0            0
ge-1/2/0.0          0            0            0            0
ge-1/2/3.0          0            0            0            0
Protocol inet, MTU: 1500, Generation: 1702, Route table: 0
  Flags: Sendbroadcast-pkt-to-re
  Addresses, Flags: Is-Preferred Is-Primary
    Destination: 10.1.1/24, Local: 10.1.1.1, Broadcast: 10.1.1.255, Generation: 1503
Protocol iso, MTU: 1497, Generation: 1703, Route table: 0
Protocol multiservice, MTU: Unlimited, Generation: 1704, Route table: 0
  Policer: Input: __default_arp_policer__

```

user@R0> **show interfaces ae2.0 extensive**

Logical interface ae2.0 (Index 337) (SNMP ifIndex 961) (Generation 827)

Flags: Up SNMP-Traps 0x4004000 Encapsulation: ENET2

Statistics	Packets	pps	Bytes	bps
------------	---------	-----	-------	-----

Bundle:

Input :	813	4	237569	3968
Output:	1786258	<b>37008</b>	196605473	32568272

Adaptive Statistics:

Adaptive Adjusts:	0
Adaptive Scans :	0
Adaptive Updates:	0

Link:

ge-0/1/8.0

Input :	163	1	20212	992
Output:	446715	9282	49138650	8168408

ge-1/1/5.0

Input :	162	1	20088	992
Output:	443846	9209	48822970	8104224

ge-1/2/8.0

Input :	161	1	19964	992
Output:	443943	9341	48833699	8220624

ge-1/3/3.0

Input :	327	1	177305	992
Output:	451754	9176	49810154	8075016

Aggregate member links: 4

```

LACP info:      Role      System      System      Port      Port      Port
                priority    identifier  priority    number    key
ge-0/1/8.0      Actor      127 3c:61:04:2f:c7:c0  127      3      3
ge-0/1/8.0      Partner    127 3c:61:04:2d:9f:c0  127      3      3
ge-1/1/5.0      Actor      127 3c:61:04:2f:c7:c0  127      5      3
ge-1/1/5.0      Partner    127 3c:61:04:2d:9f:c0  127      5      3
ge-1/2/8.0      Actor      127 3c:61:04:2f:c7:c0  127     11      3
ge-1/2/8.0      Partner    127 3c:61:04:2d:9f:c0  127     11      3
ge-1/3/3.0      Actor      127 3c:61:04:2f:c7:c0  127     12      3
ge-1/3/3.0      Partner    127 3c:61:04:2d:9f:c0  127     12      3

LACP Statistics:  LACP Rx    LACP Tx    Unknown Rx  Illegal Rx
ge-0/1/8.0        157        153         0           0
ge-1/1/5.0        160        153         0           0
ge-1/2/8.0        159        154         0           0
ge-1/3/3.0        159        154         0           0

Marker Statistics: Marker Rx    Resp Tx    Unknown Rx  Illegal Rx
ge-0/1/8.0         0          0          0           0
ge-1/1/5.0         0          0          0           0
ge-1/2/8.0         0          0          0           0
ge-1/3/3.0         0          0          0           0

Protocol inet, MTU: 1500, Generation: 1705, Route table: 0
  Flags: Sendbcast-pkt-to-re
  Addresses, Flags: Is-Preferred Is-Primary
    Destination: 10.2.1/24, Local: 10.2.1.1, Broadcast: 10.2.1.255, Generation: 1505
Protocol iso, MTU: 1497, Generation: 1706, Route table: 0
Protocol multiservice, MTU: Unlimited, Generation: 1707, Route table: 0
  Policer: Input: __default_arp_policer__

```

## Meaning

IS-IS infers that the ae0 bundle has only 3g of bandwidth available. Therefore, modifies per-packet load balancing according to the available bandwidth. As per the output, only 27 percent of the bandwidth is available on ae0 because one of the aggregated Ethernet links is down. Thus IS-IS distributes traffic unequally depending on the available bandwidth.

## Verifying Unequal Traffic Distribution on Logical Interfaces

### Purpose

To verify that IS-IS distributes traffic unevenly on logical interfaces based on the configured logical bandwidth.

## Action

```
user@R0> show interfaces ae1.1
```

```
Logical interface ae1.1 (Index 605) (SNMP ifIndex 1910)
```

```
Flags: Up SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1 ] Encapsulation: ENET2
```

```
Bandwidth: 2000mbps
```

Statistics	Packets	pps	Bytes	bps
------------	---------	-----	-------	-----

```
Bundle:
```

Input :	807	0	200537	360
---------	-----	---	--------	-----

Output:	277	0	<b>55529</b>	0
---------	-----	---	--------------	---

```
Adaptive Statistics:
```

```
Adaptive Adjusts: 0
```

```
Adaptive Scans : 0
```

```
Adaptive Updates: 0
```

```
Protocol inet, MTU: 1500
```

```
Max nh cache: 75000, New hold nh limit: 75000, Curr nh cnt: 1, Curr new hold cnt: 0, NH drop cnt: 0
```

```
Flags: Sendbcst-pkt-to-re
```

```
Addresses, Flags: Is-Preferred Is-Primary
```

```
Destination: 13.1.1/24, Local: 13.1.1.2, Broadcast: 13.1.1.255
```

```
Protocol iso, MTU: 1497
```

```
Protocol multiservice, MTU: Unlimited
```

```
user@R0> show interfaces ae1.2
```

```
Logical interface ae1.2 (Index 606) (SNMP ifIndex 1911)
```

```
Flags: Up SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.2 ] Encapsulation: ENET2
```

```
Bandwidth: 1000mbps
```

Statistics	Packets	pps	Bytes	bps
------------	---------	-----	-------	-----

```
Bundle:
```

Input :	836	0	208643	720
---------	-----	---	--------	-----

Output:	305	0	<b>61006</b>	0
---------	-----	---	--------------	---

```
Adaptive Statistics:
```

```
Adaptive Adjusts: 0
```

```
Adaptive Scans : 0
```

```
Adaptive Updates: 0
```

```
Protocol inet, MTU: 1500
```

```
Max nh cache: 75000, New hold nh limit: 75000, Curr nh cnt: 1, Curr new hold cnt: 0, NH drop cnt: 0
```

```
Flags: Sendbcst-pkt-to-re
```

```
Addresses, Flags: Is-Preferred Is-Primary
```

```
Destination: 13.2.1/24, Local: 13.2.1.2, Broadcast: 13.2.1.255
```

Protocol iso, MTU: 1497

Protocol multiservice, MTU: Unlimited

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
15.1F4	Beginning with Junos OS Release 15.1F4, you can configure the IS-IS protocol to get the logical interface bandwidth information associated with the gateways of equal-cost multipath (ECMP) next hop.

RELATED DOCUMENTATION

<i>multipath</i>
<a href="#">Understanding Weighted ECMP Traffic Distribution on One-Hop IS-IS Neighbors</a>   378

# Configuring IS-IS Traffic Engineering

## IN THIS CHAPTER

- [IS-IS Extensions to Support Traffic Engineering | 410](#)
- [Using Labeled-Switched Paths to Augment SPF to Compute IGP Shortcuts | 411](#)
- [Example: Enabling IS-IS Traffic Engineering Support | 412](#)
- [Understanding Forwarding Adjacencies | 433](#)
- [Example: Advertising Label-Switched Paths into IS-IS | 433](#)
- [Understanding Wide IS-IS Metrics for Traffic Engineering | 445](#)
- [Example: Enabling Wide IS-IS Metrics for Traffic Engineering | 446](#)
- [Understanding LDP-IGP Synchronization | 449](#)
- [Example: Configuring Synchronization Between IS-IS and LDP | 452](#)
- [Layer 2 Mapping for IS-IS | 458](#)
- [Example: Configuring Layer 2 Mapping for IS-IS | 459](#)
- [Overview of Segment Routing Statistics | 469](#)
- [Overview of Adjacency Segments, Anycast Segments, and Configurable SRGB in SPRING | 471](#)
- [Example: Configuring SRGB in Segment Routing for IS-IS | 475](#)
- [Example: Configuring Anycast and Prefix Segments in SPRING for IS-IS | 483](#)
- [Configure Segment Routing Global Blocks Label Ranges in SPRING for IS-IS Protocol | 504](#)
- [Configure Anycast and Prefix segments in SPRING for IS-IS Protocol | 506](#)
- [Flexible Algorithms in IGP for Segment Routing | 510](#)
- [Configuring Flexible Algorithm for Segment Routing | 522](#)
- [Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS and OSPF | 524](#)
- [Configuring Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS | 529](#)
- [Example: Configuring Topology Independent Loop-Free Alternate with Segment Routing for IS-IS | 531](#)
- [Static Adjacency Segment Identifier for IS-IS and OSPF | 549](#)
- [Overview of Segment Routing over RSVP Forwarding Adjacency in IS-IS | 559](#)
- [IGP Microloop Avoidance | 560](#)
- [Overview of SRv6 Network Programming in IS-IS Networks | 563](#)

- [Example: Configuring SRv6 Network Programming in IS-IS Networks | 570](#)
- [Understand SRv6 Static Segment Identifier | 601](#)
- [Binding SID \(BSID\) for Static SRv6 TE Tunnels | 605](#)
- [How to Enable Link Delay Measurement and Advertising in IGP | 607](#)
- [How to Enable Strict SPF SIDs and IGP Shortcut | 663](#)

## IS-IS Extensions to Support Traffic Engineering

### IN THIS SECTION

- [IS-IS IGP Shortcuts | 410](#)

To help provide traffic engineering and MPLS with information about network topology and loading, extensions have been added to the Junos OS implementation of IS-IS. Specifically, IS-IS supports new type, length, and value (TLV) tuples that specify link attributes. These TLVs are included in the IS-IS link-state PDUs. The link-attribute information is used to populate the traffic engineering database, which is used by the Constrained Shortest Path First (CSPF) algorithm to compute the paths that MPLS label-switched paths (LSPs) take. This path information is used by RSVP to set up LSPs and reserve bandwidth for them.



**NOTE:** Whenever possible, use IS-IS interior gateway protocol (IGP) shortcuts instead of traffic engineering shortcuts.

The traffic engineering extensions are defined in RFC 5305, *IS-IS Extensions for Traffic Engineering*.

### IS-IS IGP Shortcuts

In IS-IS, you can configure shortcuts, which allow IS-IS to use an LSP as the next hop as if it were a subinterface from the ingress routing device to the egress routing device. The address specified in the to statement at the `[edit protocols mpls label-switched-path lsp-path-name]` hierarchy level must match the router ID of the egress routing device for the LSP to function as a direct link to the egress routing device and to be used as input to IS-IS SPF calculations.



## RELATED DOCUMENTATION

[Using Labeled-Switched Paths to Augment SPF to Compute IGP Shortcuts | 411](#)

[Example: Enabling IS-IS Traffic Engineering Support | 412](#)

## Using Labeled-Switched Paths to Augment SPF to Compute IGP Shortcuts

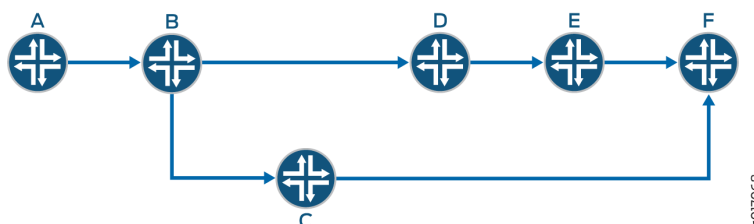
Link-state protocols, such as OSPF and IS-IS, use the shortest-path-first (SPF) algorithm to compute the shortest-path tree to all nodes in the network. The results of such computations can be represented by the destination node, next-hop address, and output interface, where the output interface is a physical interface. Label-switched paths (LSPs) can be used to augment the SPF algorithm.

IGP typically performs two independent computations. The first is performed without considering any LSP. The result of the computation is stored in the inet.0 table. This step is no different from traditional SPF computations and is always performed even if IGP shortcut is disabled.

The second computation is performed considering only LSPs as a *logical interface*. Each LSP's egress router is considered. The list of destinations whose shortest path traverses the egress router (established during the first computation) is placed in the inet.3 routing table. These destinations are given the egress router of the LSP as a next hop, enabling BGP on the local router to use these LSPs to access BGP next hops beyond the egress router. Normally, BGP can use only LSPs that terminate at the BGP next hop.

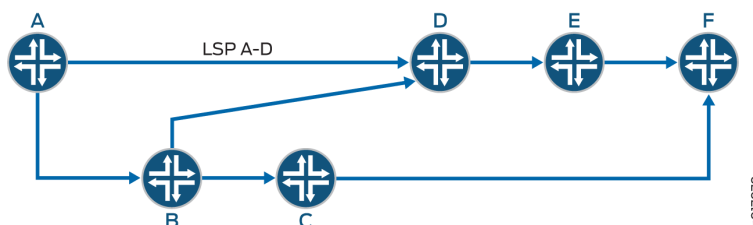
As an illustration, begin with a typical SPF tree (see [Figure 31 on page 411](#)).

**Figure 31: Typical SPF Tree, Sourced from Router A**



If an LSP connects Router A to Router D and if IGP shortcuts are enabled on Router A, you might have the SPF tree shown in [Figure 32 on page 412](#).

Figure 32: Modified SPF Tree, Using LSP A-D as a Shortcut



Note that Router D is now reachable through LSP A-D.

When computing the shortest path to reach Router D, Router A has two choices:

- Use IGP path A-B-D.
- Use LSP A-D.

Router A decides between the two choices by comparing the IGP metrics for path A-B-D with the LSP metrics for LSP A-D. If the IGP metric is lower, path A-B-D is chosen (Figure 31 on page 411). This path A-B-D is valid only when node D is not the tail-end of the LSP. If node D is the tail end of the LSP, even if the LSP metric is higher or both IGP and LSP metrics are equal, LSP A-D is used (Figure 32 on page 412).

Note that Router E is reachable through LSP A-D and Router F will take the IGP path.

## RELATED DOCUMENTATION

*traffic-engineering*

*OSPF Support for Traffic Engineering*

[IGP Shortcuts and Routing Tables](#)

## Example: Enabling IS-IS Traffic Engineering Support

### IN THIS SECTION

- Requirements | 413
- Overview | 413
- Configuration | 415

This example shows how to configure IS-IS so that it uses label-switched paths as shortcuts.

## Requirements

No special configuration beyond device initialization is required before configuring this example.

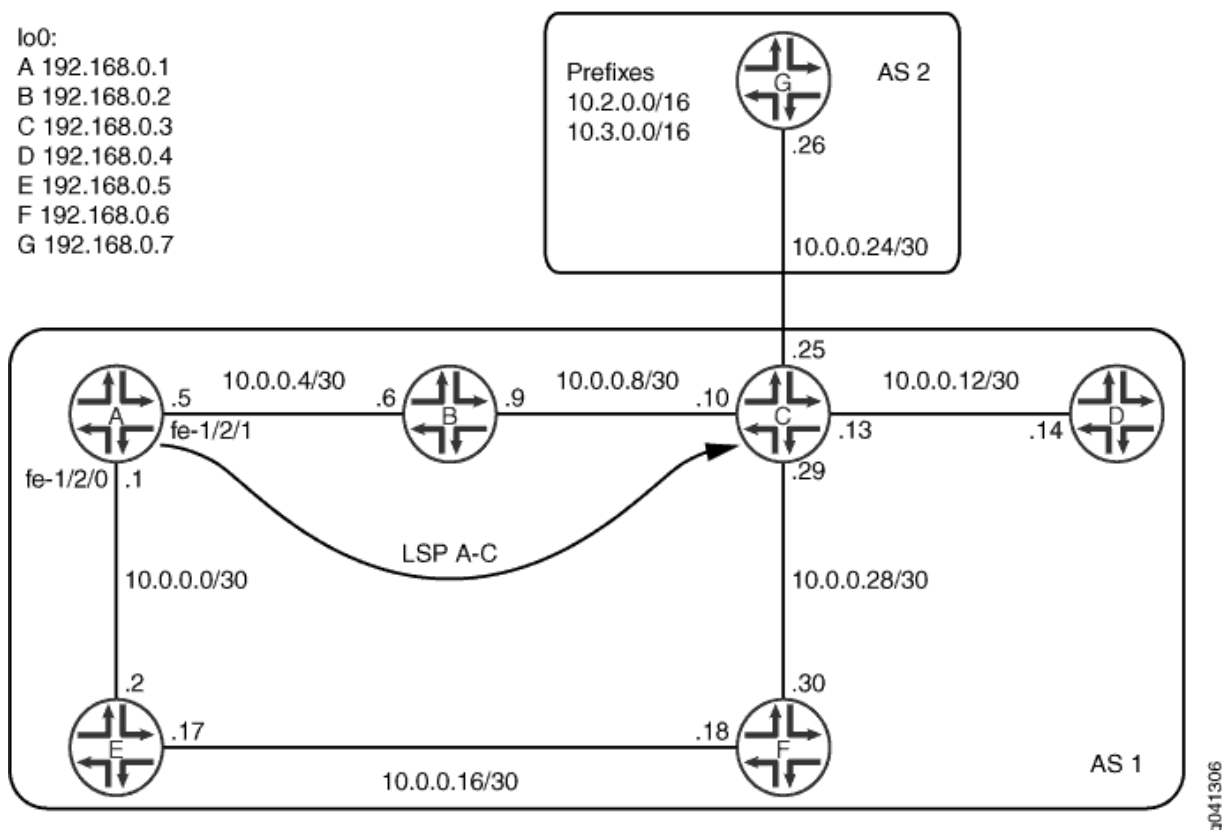
## Overview

MPLS traffic engineering maps certain data flows to established label-switched paths (LSPs) rather than to data links calculated by the interior gateway protocol (IGP) to be part of the best (shortest) path. Fundamental to this function is the determination of what traffic is to be mapped to an LSP. Traffic is mapped to an LSP at the tunnel's ingress label switching router (LSR) by designating the egress LSR as the next-hop router for certain destination prefixes.

It is important to understand that the LSP does not constitute an entire route to a destination. Rather, the LSP is a next-hop segment of the route. Therefore, packets can only be mapped to an LSP if the egress LSR is considered to be a feasible next-hop candidate during the route resolution process.

[Figure 33 on page 414](#) shows the topology used in this example.

Figure 33: IS-IS Shortcuts Topology



In this example, Device C has an external BGP (EBGP) peer session with Device G in autonomous system (AS) 2. In order to enable its internal BGP (IBGP) peers to access subnets in AS 2, Device C runs IS-IS passively on its interface connecting to Device G. IS-IS has information about the external subnets and enters routes to these subnets in the inet.0 routing table. BGP, when resolving the next-hop addresses of AS-external routes, uses the IGP route.



**TIP:** An alternative to passively running IS-IS on the interface would be to use a next-hop self policy.

Device A has an LSP to Device C. The path is configured to always go through Device E, rather than going through Device B.

Interior gateway protocol (IGP) shortcuts, also called traffic-engineering shortcuts, provide a tool by which the link-state IGP (OSPF or IS-IS) in an AS can consider an LSP in its shortest-path-first (SPF) calculations. If using passive external interfaces, the IGP views an LSP as a single data link toward the destinations beyond the LSP egress device.

When you use `traffic-engineering bgp` (which is the default) and IGP shortcuts, the traffic engineering solution is used for BGP AS-external route resolution only. However, traffic to AS-internal destinations

can also be mapped to LSPs. To accomplish this, `traffic-engineering bgp-igp` is enabled. Thus, RSVP installs the MPLS prefixes into the `inet.0` table rather than the `inet.3` table. As a result, the MPLS LSPs are installed in the forwarding table.

This approach finds practical application whenever heavy traffic is routed to specific destinations within an AS, such as server farms.

An important point about IGP shortcuts, whether used alone or in conjunction with traffic-engineering BGP-IGP, is that IGP adjacencies are never formed across the LSPs. The IGP sees the LSP as a single data link, but does not view the egress router as a potential peer and does not forward hello messages across the LSP. Also, RSVP messages are never forwarded over LSPs, preventing the possibility of an LSP being inadvertently built within another LSP.

"[CLI Quick Configuration](#)" on [page 415](#) shows the configuration for all of the devices in [Figure 33 on page 414](#). The section "[No Link Title](#)" on [page 420](#) describes the steps on Device A.

## Configuration

### IN THIS SECTION

- [Procedure](#) | [415](#)

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the `[edit]` hierarchy level.

#### Device A

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.1/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.5/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
```

```

set protocols rsvp interface lo0.0
set protocols rsvp interface fe-1/2/0.0
set protocols rsvp interface fe-1/2/1.0
set protocols mpls traffic-engineering bgp-igp
set protocols mpls label-switched-path test_path to 192.168.0.3
set protocols mpls label-switched-path test_path no-cspf
set protocols mpls label-switched-path test_path primary through_E
set protocols mpls path through_E 192.168.0.5 strict
set protocols mpls interface fe-1/2/0.0
set protocols mpls interface fe-1/2/1.0
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.1
set protocols bgp group int neighbor 192.168.0.5
set protocols bgp group int neighbor 192.168.0.6
set protocols bgp group int neighbor 192.168.0.2
set protocols bgp group int neighbor 192.168.0.3
set protocols isis traffic-engineering family inet shortcuts
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.1
set routing-options autonomous-system 1

```

## Device B

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.6/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.9/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0002.00
set protocols rsvp interface fe-1/2/0.0
set protocols rsvp interface fe-1/2/1.0
set protocols rsvp interface lo0.0
set protocols mpls interface fe-1/2/1.0
set protocols mpls interface fe-1/2/0.0
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.2
set protocols bgp group int neighbor 192.168.0.6
set protocols bgp group int neighbor 192.168.0.5

```

```

set protocols bgp group int neighbor 192.168.0.1
set protocols bgp group int neighbor 192.168.0.3
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.2
set routing-options autonomous-system 1

```

## Device C

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.10/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.13/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.25/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces fe-1/2/3 unit 0 family inet address 10.0.0.29/30
set interfaces fe-1/2/3 unit 0 family iso
set interfaces fe-1/2/3 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0003.00
set protocols rsvp interface fe-1/2/0.0
set protocols rsvp interface lo0.0
set protocols rsvp interface fe-1/2/3.0
set protocols mpls interface fe-1/2/0.0
set protocols mpls interface fe-1/2/3.0
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.3
set protocols bgp group int neighbor 192.168.0.6
set protocols bgp group int neighbor 192.168.0.5
set protocols bgp group int neighbor 192.168.0.1
set protocols bgp group int neighbor 192.168.0.2
set protocols bgp group external-peers type external
set protocols bgp group external-peers export send-some-isis
set protocols bgp group external-peers peer-as 2
set protocols bgp group external-peers neighbor 10.0.0.26
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface fe-1/2/2.0 level 1 disable
set protocols isis interface fe-1/2/2.0 level 2 passive
set protocols isis interface fe-1/2/3.0 level 1 disable

```

```

set protocols isis interface lo0.0
set policy-options policy-statement send-some-isis term 1 from protocol isis
set policy-options policy-statement send-some-isis term 1 from route-filter 10.0.0.0/24 orlonger
set policy-options policy-statement send-some-isis term 1 from route-filter 192.168.0.0/24
orlonger
set policy-options policy-statement send-some-isis term 1 then accept
set routing-options router-id 192.168.0.3
set routing-options autonomous-system 1

```

## Device D

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.14/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.4/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0004.00
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.4
set routing-options autonomous-system 1

```

## Device E

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.2/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.17/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.5/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0005.00
set protocols rsvp interface lo0.0
set protocols rsvp interface fe-1/2/0.0
set protocols rsvp interface fe-1/2/1.0
set protocols mpls interface fe-1/2/0.0
set protocols mpls interface fe-1/2/1.0
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.5
set protocols bgp group int neighbor 192.168.0.1
set protocols bgp group int neighbor 192.168.0.6
set protocols bgp group int neighbor 192.168.0.2

```



```

set protocols bgp group int neighbor 192.168.0.3
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.5
set routing-options autonomous-system 1

```

## Device F

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.18/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.30/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces fe-1/2/2 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.6/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0006.00
set protocols rsvp interface lo0.0
set protocols rsvp interface fe-1/2/0.0
set protocols rsvp interface fe-1/2/1.0
set protocols rsvp interface fe-1/2/2.0
set protocols mpls interface fe-1/2/0.0
set protocols mpls interface fe-1/2/1.0
set protocols mpls interface fe-1/2/2.0
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.6
set protocols bgp group int neighbor 192.168.0.1
set protocols bgp group int neighbor 192.168.0.5
set protocols bgp group int neighbor 192.168.0.2
set protocols bgp group int neighbor 192.168.0.3
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface fe-1/2/2.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.6
set routing-options autonomous-system 1

```

## Device G

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.26/30
set interfaces lo0 unit 0 family inet address 192.168.0.7/32 primary

```

```

set interfaces lo0 unit 0 family inet address 10.2.1.1/32
set interfaces lo0 unit 0 family inet address 10.3.1.1/32
set protocols bgp group external-peers type external
set protocols bgp group external-peers export statics
set protocols bgp group external-peers export send-directs
set protocols bgp group external-peers peer-as 1
set protocols bgp group external-peers neighbor 10.0.0.25
set policy-options policy-statement statics from protocol static
set policy-options policy-statement statics then accept
set policy-options policy-statement send-directs term 1 from protocol direct
set policy-options policy-statement send-directs term 1 then accept
set routing-options static route 10.2.0.0/32 reject
set routing-options static route 10.2.0.0/32 install
set routing-options static route 10.3.0.0/32 reject
set routing-options static route 10.3.0.0/32 install
set routing-options router-id 192.168.0.7
set routing-options autonomous-system 2

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure IS-IS traffic-engineering shortcuts:

1. Configure the interfaces.

```

[edit interfaces]
user@A# set fe-1/2/0 unit 0 family inet address 10.0.0.1/30
user@A# set fe-1/2/0 unit 0 family iso
user@A# set fe-1/2/0 unit 0 family mpls
user@A# set fe-1/2/1 unit 0 family inet address 10.0.0.5/30
user@A# set fe-1/2/1 unit 0 family iso
user@A# set fe-1/2/1 unit 0 family mpls
user@A# set lo0 unit 0 family inet address 192.168.0.1/32
user@A# set lo0 unit 0 family iso address 49.0002.0192.0168.0001.00

```

2. Enable a signaling protocol on the interfaces.

```
[edit protocols rsvp]
user@A# set interface lo0.0
user@A# set interface fe-1/2/0.0
user@A# set interface fe-1/2/1.0
```

3. Enable MPLS on the interfaces.

```
[edit protocols mpls]
user@A# set interface fe-1/2/0.0
user@A# set interface fe-1/2/1.0
```

4. Configure the label-switched path.

A single LSP, named `test_path`, is configured from Device A to Device C. The LSP explicit route object (ERO) is specified to use a strict hop through Device E, so that the LSP takes a different path from the OSPF shortest path of A-B-C. The LSP is signaled using RSVP, but no CSPF is running.

```
[edit protocols mpls]
user@A# set label-switched-path test_path to 192.168.0.3
user@A# set label-switched-path test_path no-cspf
user@A# set label-switched-path test_path primary through_E
user@A# set path through_E 192.168.0.5 strict
```

5. Configure traffic engineering for both BGP and IGP destinations.

When IGP shortcuts are also enabled, the IGP can use the LSP in its calculations. The results of the calculations are entered into the `inet.0` table.

```
[edit protocols mpls]
user@A# set traffic-engineering bgp-igp
```

6. Configure internal BGP (IBGP) peering among the devices.

```
[edit protocols bgp group int]
user@A# set type internal
user@A# set local-address 192.168.0.1
```

```

user@A# set neighbor 192.168.0.5
user@A# set neighbor 192.168.0.6
user@A# set neighbor 192.168.0.2
user@A# set neighbor 192.168.0.3

```

7. Enable IS-IS on the interfaces, and set the link metric.

```

[edit protocols isis]
user@A# set interface fe-1/2/0.0 level 1 disable
user@A# set interface fe-1/2/1.0 level 1 disable
user@A# set interface lo0.0

```

8. Configure IS-IS to use MPLS LSPs as next hops for the IPv4 address family.

It is only necessary to enable IGP shortcuts on the ingress router because that is the router performing the shortest-path-first (SPF) calculations.

It is important to understand how IGP shortcuts affect the protocol and routing table relationship. The IGP performs SPF calculations to subnets downstream of LSP egress points, but the results of these calculations are entered into the inet.3 table only. At the same time, the IGP performs its traditional SPF calculations and enters the results of these calculations into the inet.0 table. The result is that although the IGP is making entries into the inet.3 table, BGP is still the only protocol with visibility into that table for the purposes of route resolution. Therefore, forwarding to AS-internal destinations still uses the inet.0 IGP routes, and the LSPs are only used for BGP next-hop resolution. If you want the LSPs to be used for IGP next-hop resolution, you must configure traffic-engineering bgp-igp.

```

[edit protocols isis]
user@A# set traffic-engineering family inet shortcuts

```

9. Configure the router ID and the autonomous system (AS) number.

```

[edit routing-options]
user@A# set router-id 192.168.0.1
user@A# set autonomous-system 1

```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@A# show interfaces
fe-1/2/0 {
  unit 0{
    family inet {
      address 10.0.0.1/30;
    }
    family iso;
    family mpls;
  }
}
fe-1/2/1{
  unit 0
    family inet {
      address 10.0.0.5/30;
    }
    family iso;
    family mpls;
  }
}
lo0 {
  unit 0{
    family inet {
      address 192.168.0.1/32;
    }
    family iso {
      address 49.0002.0192.0168.0001.00;
    }
  }
}
```

```
user@A# show protocols
rsvp {
  interface lo0.0;
  interface fe-1/2/0.0;
  interface fe-1/2/1.0;
```

```

}
mpls {
    traffic-engineering bgp-igp;
    label-switched-path test_path {
        to 192.168.0.3;
        no-cspf;
        primary through_E;
    }
    path through_E {
        192.168.0.5 strict;
    }
    interface fe-1/2/0.0;
    interface fe-1/2/1.0;
}
bgp {
    group int {
        type internal;
        local-address 192.168.0.1;
        neighbor 192.168.0.5;
        neighbor 192.168.0.6;
        neighbor 192.168.0.2;
        neighbor 192.168.0.3;
    }
}
isis {
    traffic-engineering {
        family inet {
            shortcuts;
        }
    }
    interface fe-1/2/0.0 {
        level 1 disable;
    }
    interface fe-1/2/1.0 {
        level 1 disable;
    }
}

```

```
interface lo0.0;
}
```

```
user@A# show routing-options
router-id 192.168.0.1;
autonomous-system 1;
```

If you are done configuring the device, enter **commit** from configuration mode.

## Verification

### IN THIS SECTION

- [Verifying the Next Hops | 425](#)
- [Checking the RSVP Sessions | 428](#)
- [Checking the Paths with Different Traffic Engineering Settings | 430](#)

Confirm that the configuration is working properly.

### Verifying the Next Hops

#### Purpose

Verify that the MPLS LSP is used as the next hop in the expected routes.

#### Action

From operational mode, enter the `show route` command.

```
user@A> show route

inet.0: 21 destinations, 21 routes (21 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/30      *[Direct/0] 4d 09:07:26
                 >    via fe-1/2/0.0
10.0.0.1/32     *[Local/0] 4d 09:07:26
```

```

                Local via fe-1/2/0.0
10.0.0.4/30      *[Direct/0] 4d 09:07:28
                  > via fe-1/2/1.0
10.0.0.5/32      *[Local/0] 4d 09:07:28
                  Local via fe-1/2/1.0
10.0.0.8/30      *[IS-IS/18] 01:42:24, metric 20
                  > to 10.0.0.6 via fe-1/2/1.0
10.0.0.12/30     *[IS-IS/18] 01:42:24, metric 30
                  > to 10.0.0.6 via fe-1/2/1.0
10.0.0.16/30     *[IS-IS/18] 01:42:24, metric 20
                  > to 10.0.0.2 via fe-1/2/0.0
10.0.0.20/30     *[IS-IS/18] 01:42:24, metric 30
                  > to 10.0.0.2 via fe-1/2/0.0
10.0.0.24/30     *[IS-IS/18] 01:42:24, metric 30
                  > to 10.0.0.6 via fe-1/2/1.0
10.0.0.28/30     *[IS-IS/18] 01:42:24, metric 30
                  to 10.0.0.6 via fe-1/2/1.0
                  > to 10.0.0.2 via fe-1/2/0.0
10.2.0.0/32      *[BGP/170] 02:22:30, localpref 100, from 192.168.0.3
                  AS path: 2 I, validation-state: unverified
                  > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path
10.2.1.1/32      *[BGP/170] 02:20:23, localpref 100, from 192.168.0.3
                  AS path: 2 I, validation-state: unverified
                  > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path
10.3.0.0/32      *[BGP/170] 02:22:30, localpref 100, from 192.168.0.3
                  AS path: 2 I, validation-state: unverified
                  > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path
10.3.1.1/32      *[BGP/170] 02:20:23, localpref 100, from 192.168.0.3
                  AS path: 2 I, validation-state: unverified
                  > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path
192.168.0.1/32   *[Direct/0] 4d 09:08:47
                  > via lo0.0
192.168.0.2/32   *[IS-IS/18] 01:42:24, metric 10
                  > to 10.0.0.6 via fe-1/2/1.0
192.168.0.3/32   *[IS-IS/18] 01:42:24, metric 20
                  > to 10.0.0.6 via fe-1/2/1.0
192.168.0.4/32   *[IS-IS/18] 01:42:24, metric 30
                  > to 10.0.0.6 via fe-1/2/1.0
                  to 10.0.0.2 via fe-1/2/0.0
192.168.0.5/32   *[IS-IS/18] 01:42:24, metric 10
                  > to 10.0.0.2 via fe-1/2/0.0
192.168.0.6/32   *[IS-IS/18] 01:42:24, metric 20
                  > to 10.0.0.2 via fe-1/2/0.0

```



```

192.168.0.7/32      *[BGP/170] 02:20:23, localpref 100, from 192.168.0.3
                   AS path: 2 I, validation-state: unverified
                   > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path

```

inet.3: 5 destinations, 6 routes (5 active, 0 holddown, 0 hidden)

+ = Active Route, - = Last Active, \* = Both

```

10.0.0.12/30       *[IS-IS/18] 01:41:21, metric 30
                   > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path
10.0.0.24/30       *[IS-IS/18] 01:41:21, metric 30
                   > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path
10.0.0.28/30       *[IS-IS/18] 01:41:21, metric 30
                   > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path
192.168.0.3/32     *[RSVP/7/1] 01:41:21, metric 20
                   > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path
                   [IS-IS/18] 01:41:21, metric 20
                   > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path
192.168.0.4/32     *[IS-IS/18] 01:41:21, metric 30
                   > to 10.0.0.2 via fe-1/2/0.0, label-switched-path test_path

```

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

+ = Active Route, - = Last Active, \* = Both

```

49.0002.0192.0168.0001/72
                   *[Direct/0] 4d 09:08:47
                   > via lo0.0

```

mpls.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)

+ = Active Route, - = Last Active, \* = Both

```

0                  *[MPLS/0] 4d 09:10:00, metric 1
                   Receive
1                  *[MPLS/0] 4d 09:10:00, metric 1
                   Receive
2                  *[MPLS/0] 4d 09:10:00, metric 1
                   Receive
13                 *[MPLS/0] 4d 09:10:00, metric 1
                   Receive

```

## Meaning

IS-IS chooses the LSP as the shortest path to destinations downstream of the LSP egress device. Additionally, because the IGP uses the LSP to reach external subnet 10.0.0.24/30, BGP also uses the LSP in its routes to 10.2.0.0 and 10.3.0.0.

If next-hop self were used at Device C, BGP would still choose the LSP over the IGP path.

## Checking the RSVP Sessions

### Purpose

Display information about RSVP sessions

### Action

From operational mode, enter the `show rsvp session brief` command.

```
user@A> show rsvp session brief
Ingress RSVP: 1 sessions
To          From          State  Rt Style Labelin Labelout LSPname
192.168.0.3  192.168.0.1  Up     0  1 FF      -   299776 test_path
Total 1 displayed, Up 1, Down 0

Egress RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

```
user@E> show rsvp session brief
Ingress RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0

Egress RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit RSVP: 1 sessions
To          From          State  Rt Style Labelin Labelout LSPname
```

```
192.168.0.3    192.168.0.1    Up        0  1 FF  299776   299808 test_path
Total 1 displayed, Up 1, Down 0
```

```
user@F> show rsvp session brief
```

```
Ingress RSVP: 0 sessions
```

```
Total 0 displayed, Up 0, Down 0
```

```
Egress RSVP: 0 sessions
```

```
Total 0 displayed, Up 0, Down 0
```

```
Transit RSVP: 1 sessions
```

To	From	State	Rt	Style	Labelin	Labelout	LSPname
192.168.0.3	192.168.0.1	Up	0	1 FF	299808	3	test_path

```
Total 1 displayed, Up 1, Down 0
```

```
user@C> show rsvp session brief
```

```
Ingress RSVP: 0 sessions
```

```
Total 0 displayed, Up 0, Down 0
```

```
Egress RSVP: 1 sessions
```

To	From	State	Rt	Style	Labelin	Labelout	LSPname
192.168.0.3	192.168.0.1	Up	0	1 FF		3	- test_path

```
Total 1 displayed, Up 1, Down 0
```

```
Transit RSVP: 0 sessions
```

```
Total 0 displayed, Up 0, Down 0
```

## Meaning

On all four routing devices, the ingress and egress IP addresses of the LSP are shown. The path is shown as an ingress path at Device A, and packets forwarded on the LSP are assigned a label of 299776. At Device E, the LSP is transit, and packets arriving with a label of 299776 are given an outgoing label of 299808. The labels have significance only between neighboring label-switched routers (LSRs). Device F swaps incoming label 299808 for outgoing label 3. Device C, the egress, pops label 3 and routes the received packet by standard IP longest-match route lookup.

## Checking the Paths with Different Traffic Engineering Settings

### Purpose

Check the paths used for IGP and BGP routes when traffic-engineering bgp-igp is used and when traffic-engineering bgp (the default) is used.

### Action

#### 1. Configure traffic-engineering bgp.

This removes traffic-engineering bgp-igp from the configuration because only one MPLS traffic engineering setting can be configured in each routing instance.

```
[edit protocols mpls]
user@A# set traffic-engineering bgp
user@A# commit
```

#### 2. Use the show route forwarding-table command to check the paths when traffic-engineering bgp (the default) is configured.

```
user@A> show route forwarding-table destination 10.2.1.1
Routing table: default.inet
Internet:
Destination      Type RtRef Next hop          Type Index NhRef Netif
10.2.1.1/32      user   0           10.0.0.2          indr 262145   6
                  Push 299776 1013   2 fe-1/2/0.0
```

```
user@A> show route forwarding-table destination 192.168.0.3
Routing table: default.inet
Internet:
Destination      Type RtRef Next hop          Type Index NhRef Netif
192.168.0.3/32   user   1 10.0.0.6          ucst  938   11 fe-1/2/1.0
```

#### 3. Use the traceroute command to check the paths when traffic-engineering bgp (the default) is configured.

```
user@A> traceroute 10.2.1.1
traceroute to 10.2.1.1 (10.2.1.1), 30 hops max, 40 byte packets
```

```

1  10.0.0.2 (10.0.0.2)  11.086 ms  1.587 ms  1.603 ms
   MPLS Label=299776 CoS=0 TTL=1 S=1
2  10.0.0.18 (10.0.0.18)  1.455 ms  1.477 ms  1.442 ms
   MPLS Label=299808 CoS=0 TTL=1 S=1
3  10.0.0.29 (10.0.0.29)  2.240 ms  1.045 ms  1.243 ms
4  10.2.1.1 (10.2.1.1)  1.363 ms  1.389 ms  1.374 ms

```

```

user@A> traceroute 192.168.0.3
traceroute to 192.168.0.3 (192.168.0.3), 30 hops max, 40 byte packets
1  10.0.0.6 (10.0.0.6)  1.759 ms  1.872 ms  2.281 ms
2  bb03-cclab-lo0.spglab.juniper.net (192.168.0.3)  2.119 ms  2.157 ms  1.598 ms

```

#### 4. Configure traffic-engineering bgp-igp.

This removes traffic-engineering bgp from the configuration because only one MPLS traffic engineering setting can be configured in each routing instance.

```

[edit protocols mpls]
user@A# set traffic-engineering bgp-igp
user@A# commit

```

#### 5. Use the show route forwarding-table command to check the paths when traffic-engineering bgp-igp is configured.

```

user@A> show route forwarding-table destination 10.2.1.1
Routing table: default.inet
Internet:

```

Destination	Type	RtRef	Next hop	Type	Index	NhRef	Netif
10.2.1.1/32	user	0		indr	262145	6	
			10.0.0.2	<b>Push 299776</b>		1013	2 fe-1/2/0.0

```

user@A> show route forwarding-table destination 192.168.0.3
Routing table: default.inet
Internet:

```

Destination	Type	RtRef	Next hop	Type	Index	NhRef	Netif
192.168.0.3/32	user	1	10.0.0.2	<b>Push 299776</b>		1013	8 fe-1/2/0.0

6. Use the `traceroute` command to check the paths when traffic-engineering bgp-igp is configured.

```
user@A> traceroute 10.2.1.1
traceroute to 10.2.1.1 (10.2.1.1), 30 hops max, 40 byte packets
 1  10.0.0.2 (10.0.0.2)  2.348 ms  1.475 ms  1.434 ms
    MPLS Label=299776 CoS=0 TTL=1 S=1
 2  10.0.0.18 (10.0.0.18)  1.507 ms  2.307 ms  1.911 ms
    MPLS Label=299808 CoS=0 TTL=1 S=1
 3  10.0.0.29 (10.0.0.29)  1.743 ms  1.645 ms  1.940 ms
 4  10.2.1.1 (10.2.1.1)  2.041 ms  1.977 ms  2.233 ms
```

```
user@A> traceroute 192.168.0.3
traceroute to 192.168.0.3 (192.168.0.3), 30 hops max, 40 byte packets
 1  10.0.0.2 (10.0.0.2)  1.721 ms  2.558 ms  2.229 ms
    MPLS Label=299776 CoS=0 TTL=1 S=1
 2  10.0.0.18 (10.0.0.18)  2.505 ms  1.462 ms  1.408 ms
    MPLS Label=299808 CoS=0 TTL=1 S=1
 3  bb03-cclab-lo0.spglab.juniper.net (192.168.0.3)  1.371 ms  1.422 ms  1.351 ms
```

## Meaning

When traffic-engineering bgp is configured, the first trace is to a destination belonging to the BGP-learned 10.2.0.0/16 prefix, and follows the LSP. The second trace is to the IS-IS-learned 192.168.0.3 route (Device C's loopback interface address), and follows the IS-IS route. These results correspond to what we observe in the forwarding table. The forwarding table is built based on routes in inet.0 only. BGP can look into inet.3 and select an LSP as the best path to the next hop of a BGP prefix, and can add a route into inet.0 utilizing that LSP. An entry is then made to the forwarding table from the inet.0 route. No other protocol, by default, can consult inet.3, and the inet.3 routes are not entered into inet.0. Therefore, the forwarding entry for 192.168.0.3 is created from the only route to that destination in inet.0: the IS-IS route.

When traffic-engineering bgp-igp is configured, the first trace to 10.2.1.1 continues to follow the LSP. The second trace to 192.168.0.3 also follows the LSP. These results correspond to what we observe in the forwarding table, which shows that the LSP is used for IGP next-hop resolution.

## RELATED DOCUMENTATION

[Advertising LSPs into IGP](#)

*Example: Enabling OSPF Traffic Engineering Support*

## Understanding Forwarding Adjacencies

A forwarding adjacency is a traffic engineering label-switched path (LSP) that is configured between two nodes and that is used by the interior gateway protocol (IGP) to forward traffic.

When you set up MPLS traffic-engineering tunnels between sites, by default the IGP does not consider those tunnels for traffic forwarding. Forwarding adjacencies allow you to treat a traffic engineering LSP tunnel as a link in an IGP topology. The link is used in the shortest-path-first (SPF) algorithm and is advertised to the IGP peers. A forwarding adjacency can be created between routing devices regardless of their location in the network.

### RELATED DOCUMENTATION

[Example: Advertising Label-Switched Paths into IS-IS | 433](#)

## Example: Advertising Label-Switched Paths into IS-IS

### IN THIS SECTION

- [Requirements | 433](#)
- [Overview | 434](#)
- [Configuration | 434](#)
- [Verification | 442](#)

This example shows how to advertise label-switched paths (LSPs) into IS-IS as point-to-point links (sometimes referred to as forwarding adjacencies) so that the LSPs can be used in SPF calculations. The advertisement contains a local address (the **from** address of the LSP), a remote address (the **to** address of the LSP), and a metric.

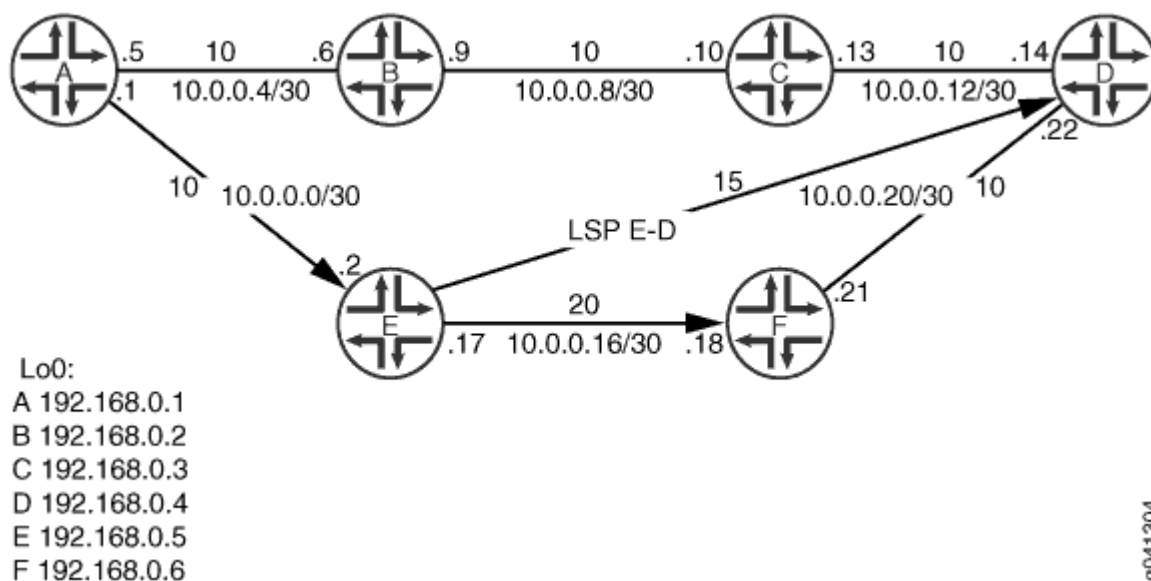
### Requirements

No special configuration beyond device initialization is required before configuring this example.

## Overview

Figure 34 on page 434 shows the topology used in this example.

Figure 34: IS-IS Advertising a Label-Switched Path Topology



The example shows how to configure the LSP from Device E to Device D and then advertise this path through IS-IS. The configuration is verified by performing a traceroute operation from Device A to Device D and making sure that the LSP is used for forwarding.

"CLI Quick Configuration" on page 435 shows the configuration for all of the devices in Figure 34 on page 434. The section "No Link Title" on page 438 describes the steps on Device E.

## Configuration

### IN THIS SECTION

- Procedure | 435



## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device A

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.1/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.5/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set protocols rsvp interface lo0.0
set protocols rsvp interface fe-1/2/0.0
set protocols mpls interface fe-1/2/0.0
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.1
set protocols bgp group int neighbor 192.168.0.6
set protocols bgp group int neighbor 192.168.0.5
set protocols bgp group int neighbor 192.168.0.4
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/0.0 level 2 metric 10
set protocols isis interface fe-1/2/0.5 level 2 metric 10
set protocols isis interface fe-1/2/0.5 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.1
set routing-options autonomous-system 1
```

#### Device B

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.6/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.9/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0002.00
set protocols isis interface fe-1/2/0.0 level 1 disable
```

```

set protocols isis interface fe-1/2/1.0 level 2 metric 10
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.2

```

### Device C

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.10/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.13/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0003.00
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 2 metric 10
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.3

```

### Device D

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.14/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.22/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.4/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0004.00
set protocols rsvp interface lo0.0
set protocols rsvp interface fe-1/2/1.0
set protocols mpls label-switched-path E-D to 192.168.0.5
set protocols mpls interface fe-1/2/1.0
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.4
set protocols bgp group int neighbor 192.168.0.5
set protocols bgp group int neighbor 192.168.0.1
set protocols bgp group int neighbor 192.168.0.6
set protocols isis interface fe-1/2/0.14 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0
set protocols isis label-switched-path E-D level 2

```

```
set routing-options router-id 192.168.0.4
set routing-options autonomous-system 1
```

## Device E

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.2/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.17/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.5/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0005.00
set protocols rsvp interface lo0.0
set protocols rsvp interface fe-1/2/0.0
set protocols rsvp interface fe-1/2/1.0
set protocols mpls label-switched-path E-D to 192.168.0.4
set protocols mpls interface fe-1/2/0.0
set protocols mpls interface fe-1/2/1.0
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.5
set protocols bgp group int neighbor 192.168.0.6
set protocols bgp group int neighbor 192.168.0.1
set protocols bgp group int neighbor 192.168.0.4
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 2 metric 20
set protocols isis interface lo0.0
set protocols isis label-switched-path E-D level 2 metric 15
set routing-options router-id 192.168.0.5
set routing-options autonomous-system 1
```

## Device F

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.18/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family mpls
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.21/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.6/32
```

```

set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0006.00
set protocols rsvp interface lo0.0
set protocols rsvp interface fe-1/2/0.0
set protocols rsvp interface fe-1/2/1.0
set protocols mpls interface fe-1/2/0.0
set protocols mpls interface fe-1/2/1.0
set protocols bgp group int type internal
set protocols bgp group int local-address 192.168.0.6
set protocols bgp group int neighbor 192.168.0.1
set protocols bgp group int neighbor 192.168.0.5
set protocols bgp group int neighbor 192.168.0.4
set protocols isis interface fe-1/2/0.0 level 1 disable
set protocols isis interface fe-1/2/1.0 level 2 metric 10
set protocols isis interface fe-1/2/1.0 level 1 disable
set protocols isis interface lo0.0
set routing-options router-id 192.168.0.6
set routing-options autonomous-system 1

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To advertise LSPs into IS-IS:

1. Configure the interfaces.

```

[edit interfaces]
user@E# set fe-1/2/0 unit 0 family inet address 10.0.0.2/30
user@E# set fe-1/2/0 unit 0 family iso
user@E# set fe-1/2/0 unit 0 family mpls
user@E# set fe-1/2/1 unit 0 family inet address 10.0.0.17/30
user@E# set fe-1/2/1 unit 0 family iso
user@E# set fe-1/2/1 unit 0 family mpls
user@E# set lo0 unit 0 family inet address 192.168.0.5/32
user@E# set lo0 unit 0 family iso address 49.0002.0192.0168.0005.00

```

2. Enable a signaling protocol on the interfaces.

```
[edit protocols rsvp]
user@E# set interface lo0.0
user@E# set interface fe-1/2/0.0
user@E# set interface fe-1/2/1.0
```

3. Enable MPLS on the interfaces.

```
[edit protocols mpls]
user@E# set interface fe-1/2/0.0
user@E# set interface fe-1/2/1.0
```

4. Configure the LSP.

Make sure that you configure the reverse LSP on the endpoint, in this case on Device D.

```
[edit protocols mpls]
user@E# set label-switched-path E-D to 192.168.0.4
```

5. Configure internal BGP (IBGP) peering among the devices that must run MPLS.

```
[edit protocols bgp group int]
user@E# set type internal
user@E# set local-address 192.168.0.5
user@E# set neighbor 192.168.0.6
user@E# set neighbor 192.168.0.1
user@E# set neighbor 192.168.0.4
```

6. Enable IS-IS on the interfaces, and set the link metric.

IS-IS Level 1 and Level 2 are enabled when you include the interface at [edit protocols isis]. By disabling Level 1, you are in effect creating a Level 2 IS-IS interface.

```
[edit protocols isis]
user@E# set interface fe-1/2/0.0 level 1 disable
user@E# set interface fe-1/2/1.0 level 1 disable
```

```
user@E# set interface fe-1/2/1.0 level 2 metric 20
user@E# set interface lo0.0
```

## 7. Advertise the LSP through IS-IS.

Make sure that you advertise the LSP on the endpoint, in this case on Device D.

```
[edit protocols isis]
user@E# set label-switched-path E-D level 2 metric 15
```

## 8. Configure the router ID and the autonomous system (AS) number.

```
[edit routing-options]
user@E# set router-id 192.168.0.5
user@E# set autonomous-system 1
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@E# show interfaces
fe-1/2/0 {
  unit 0 {
    family inet {
      address 10.0.0.2/30;
    }
    family iso;
    family mpls;
  }
}
fe-1/2/1 {
  unit 0 {
    family inet {
      address 10.0.0.17/30;
    }
    family iso;
    family mpls;
  }
}
```

```

}
lo0 {
    unit 0 {
        family inet {
            address 192.168.0.5/32;
        }
        family iso {
            address 49.0002.0192.0168.0005.00;
        }
    }
}
}

```

```

user@E# show protocols
rsvp {
    interface lo0.0;
    interface fe-1/2/0.0;
    interface fe-1/2/1.0;
}
mpls {
    label-switched-path E-D {
        to 192.168.0.4;
    }
    interface fe-1/2/0.0;
    interface fe-1/2/1.0;
}
bgp {
    group int {
        type internal;
        local-address 192.168.0.5;
        neighbor 192.168.0.6;
        neighbor 192.168.0.1;
        neighbor 192.168.0.4;
    }
}
isis {
    interface fe-1/2/0.0 {
        level 1 disable;
    }
    interface fe-1/2/1.0 {
        level 1 disable;
        level 2 metric 20;
    }
}

```

```

    }
    interface lo0.0;
    label-switched-path E-D {
        level 2 metric 15;
    }
}

```

```

user@E# show routing-options
router-id 192.168.0.5;
autonomous-system 1;

```

If you are done configuring the device, enter **commit** from configuration mode.

## Verification

### IN THIS SECTION

- [Verifying the IS-IS Neighbor | 442](#)
- [Checking the IS-IS SPF Calculations | 443](#)
- [Checking the Forwarding Path | 444](#)

Confirm that the configuration is working properly.

### Verifying the IS-IS Neighbor

#### Purpose

Verify that another neighbor is listed and is reachable over the LSP. The interface field indicates the name of the LSP.

#### Action

From operational mode, enter the `show isis adjacency detail` command.

```

user@E> show isis adjacency detail
D
  Interface: E-D, Level: 2, State: One-way, Expires in 0 secs

```



Priority: 0, Up/Down transitions: 1, Last transition: 1d 00:34:58 ago  
 Circuit type: 3, Speaks: IP  
 Topologies: Unicast  
 Restart capable: No, Adjacency advertisement: Advertise  
 IP addresses: 192.168.0.4

F

Interface: fe-1/2/1.0, Level: 2, State: Up, Expires in 7 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 1d 01:16:22 ago  
 Circuit type: 2, Speaks: IP, IPv6, MAC address: 0:5:85:8f:c8:bd  
 Topologies: Unicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: F.02, IP addresses: 10.0.0.18

A

Interface: fe-1/2/0.0, Level: 2, State: Up, Expires in 20 secs  
 Priority: 64, Up/Down transitions: 1, Last transition: 1d 01:17:20 ago  
 Circuit type: 2, Speaks: IP, IPv6, MAC address: 0:5:85:8f:c8:bc  
 Topologies: Unicast  
 Restart capable: Yes, Adjacency advertisement: Advertise  
 LAN id: E.02, IP addresses: 10.0.0.1

## Meaning

As expected, Interface: E-D is shown in the output, and the state is shown as One-way.

## Checking the IS-IS SPF Calculations

### Purpose

Verify that the LSP is being used in the SPF calculations.

### Action

From operational mode, enter the `show isis spf brief` command.

```
user@E> show isis spf brief
```

IS-IS level 1 SPF results:

Node	Metric	Interface	NH	Via	SNPA
------	--------	-----------	----	-----	------

```

E.00      0
  1 nodes

IS-IS level 2 SPF results:
Node      Metric    Interface    NH    Via    SNPA
C.02      30        fe-1/2/0.0    IPV4 A      0:5:85:8f:c8:bc
C.00      25        fe-1/2/1.0    LSP  E-D
D.03      25        fe-1/2/1.0    LSP  E-D
D.02      25        fe-1/2/1.0    LSP  E-D
F.00      20        fe-1/2/1.0    IPV4 F      0:5:85:8f:c8:bd
B.00      20        fe-1/2/0.0    IPV4 A      0:5:85:8f:c8:bc
B.02      20        fe-1/2/0.0    IPV4 A      0:5:85:8f:c8:bc
D.00      15        fe-1/2/1.0    LSP  E-D
A.00      10        fe-1/2/0.0    IPV4 A      0:5:85:8f:c8:bc
E.02      10
E.00      0
  11 nodes

```

## Meaning

As expected, the SPF results include the LSP, E-D.

## Checking the Forwarding Path

## Purpose

Verify that a traceroute operation from Device A to Device D uses the LSP.

## Action

```

user@A> traceroute 192.168.0.4
traceroute to 192.168.0.4 (192.168.0.4), 30 hops max, 40 byte packets
 1  10.0.0.2 (10.0.0.2)  1.092 ms  1.034 ms  1.174 ms
 2  10.0.0.18 (10.0.0.18)  1.435 ms  2.062 ms  2.232 ms
    MPLS Label=299792 CoS=0 TTL=1 S=1
 3  bb04-cclab-lo0.spglab.juniper.net (192.168.0.4)  2.286 ms  1.432 ms  1.354 ms

```

## Meaning

The output shows that the LSP is used.

## RELATED DOCUMENTATION

| [Advertising LSPs into IGPs](#)

## Understanding Wide IS-IS Metrics for Traffic Engineering

All OSPF and IS-IS interfaces have a cost, which is a routing metric that is used in the link-state calculation. Routes with lower total path metrics are preferred over those with higher path metrics. Unlike OSPF, in which the link metric is calculated automatically based on bandwidth, there is no automatic calculation for IS-IS. All IS-IS links use a metric of 10 by default.

Normally, IS-IS metrics can have values up to 63. The total cost to a destination is the sum of the metrics on all outgoing interfaces along a particular path from the source to the destination. By default, the total path metric is limited to 1023. This metric value is insufficient for large networks and provides too little granularity for traffic engineering, especially with high-bandwidth links. A wider range of metrics is also required if route leaking is used.

IS-IS generates two type, length, and value (TLV) tuples, one for an IS-IS adjacency and the second for an IP prefix. To allow IS-IS to support traffic engineering, a second pair of TLVs has been added to IS-IS, one for IP prefixes and the second for IS-IS adjacency and traffic engineering information. With these TLVs, IS-IS metrics can have values up to 16,777,215 ( $2^{24} - 1$ ).

By default, Junos OS supports the sending and receiving of wide metrics. Junos OS allows a maximum metric value of 63 and generates both pairs of TLVs. To configure IS-IS to generate only the new pair of TLVs and thus to allow the wider range of metric values, you must include the `wide-metrics-only` statement in the IS-IS configuration.

The combination of `wide-metrics-only` and `traffic-engineering disable` configuration options under IS-IS protocols suppresses the combination of the TLVs 2, 22, 128, 134, and 135 IS-IS routing information for that level. That means that the local server will not send the TLVs but accepts them when received. The effect of the configuration options on TLVs 2, 22, 128, 134, and 135 will be individually evaluated.

See [[Understanding the effects of ISIS wide-metric-only and traffic-engineering disable configuration options in Junos.](#)]

## RELATED DOCUMENTATION

| [Example: Enabling Wide IS-IS Metrics for Traffic Engineering](#) | 446

## Example: Enabling Wide IS-IS Metrics for Traffic Engineering

### IN THIS SECTION

- [Requirements | 446](#)
- [Overview | 446](#)
- [Configuration | 446](#)
- [Verification | 448](#)

This example shows how to allow a wide range of metric values on IS-IS interfaces.

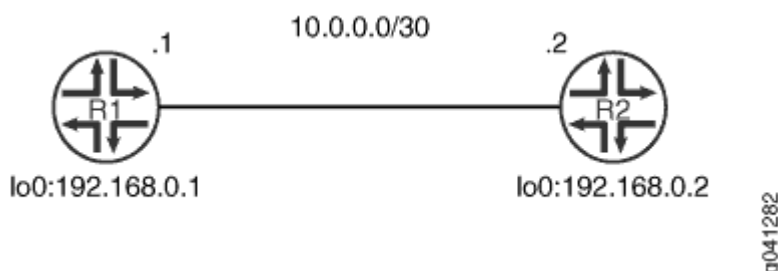
### Requirements

Before you begin, configure IS-IS on both routers. See ["Example: Configuring IS-IS" on page 15](#) for information about the sample IS-IS configuration.

### Overview

[Figure 35 on page 446](#) shows the topology used in this example.

**Figure 35: IS-IS Wide Metrics Topology**



This example describes the steps on Device R1.

### Configuration

#### IN THIS SECTION

- [Procedure | 447](#)

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R1

```
set protocols isis level 2 wide-metrics-only
set protocols isis level 1 wide-metrics-only
set protocols isis interface lt-1/2/0.1 level 2 metric 100
set protocols isis interface lt-1/2/0.1 level 1 metric 100
```

### Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure IS-IS checksums:

1. Configure a metric of 100 on the interface at both IS-IS levels.

```
[edit protocols isis interface lt-1/2/0.1]
user@R1# set level 2 metric 100
user@R1# set level 1 metric 100
```

2. Enable wide metrics.

```
[edit protocols isis]
user@R1# set level 2 wide-metrics-only
user@R1# set level 1 wide-metrics-only
```

## Results

From configuration mode, confirm your configuration by entering the `show protocols` command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show protocols
isis {
  level 2 wide-metrics-only;
  level 1 wide-metrics-only;
  interface lt-1/2/0.1 {
    level 2 metric 100;
    level 1 metric 100;
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

## Verification

### IN THIS SECTION

- [Verifying That Wide Metrics Are Enabled | 448](#)

Confirm that the configuration is working properly.

### Verifying That Wide Metrics Are Enabled

#### Purpose

Make sure that the interface has the expected metric.

#### Action

From operational mode, enter the `show isis interface extensive` command.

```
user@R1> show isis interface lt-1/2/0.1 extensive
IS-IS interface database:
```

```

lt-1/2/0.1
  Index: 68, State: 0x6, Circuit id: 0x1, Circuit type: 3
  LSP interval: 100 ms, CSNP interval: 10 s, Loose Hello padding
  Adjacency advertisement: Advertise
  Level 1
    Adjacencies: 1, Priority: 64, Metric: 100
    Hello Interval: 9.000 s, Hold Time: 27 s
    Designated Router: tp5-R2.02 (not us)
  Level 2
    Adjacencies: 1, Priority: 64, Metric: 100
    Hello Interval: 9.000 s, Hold Time: 27 s
    Designated Router: tp5-R2.02 (not us)

```

## Meaning

The output shows that the metric is set to 100, as expected, at both Level 1 and Level 2.

## RELATED DOCUMENTATION

[Understanding Hitless Authentication Key Rollover for IS-IS | 84](#)

[Example: Configuring Hitless Authentication Key Rollover for IS-IS | 86](#)

## Understanding LDP-IGP Synchronization

### IN THIS SECTION

- [Synchronization Behavior During Graceful Restart | 451](#)
- [Synchronization Behavior on LAN Interfaces | 451](#)
- [Synchronization Behavior on IGP Passive Interfaces | 451](#)
- [Synchronization and TE Metrics | 452](#)

Synchronization between the Label Distribution Protocol (LDP) and the underlying interior gateway protocol (IGP) ensures that LDP is fully established before the IGP path is used for forwarding traffic.

LDP is often used to establish MPLS label-switched paths (LSPs) throughout a complete network domain using an IGP such as OSPF or IS-IS. In such a network, all links in the domain have IGP adjacencies as well as LDP adjacencies. LDP establishes the LSPs on the shortest path to a destination as determined by IP forwarding.

If the IGP and LDP are not synchronized, packet loss can occur. This issue is especially significant for applications such as a core network that does not employ BGP. Another example is an MPLS VPN where each provider edge (PE) router depends on the availability of a complete MPLS forwarding path to the other PE devices for each VPN that it serves. This means that along the shortest path between the PE routers, each link must have an operational hello adjacency and an operational LDP session, and MPLS label bindings must have been exchanged over each session.

LDP establishes MPLS LSPs along the shortest path to the destination as determined by IP forwarding. In a Layer 2 VPN or Layer 3 VPN scenario, if the LSP is not yet formed between the PE devices, services depending on MPLS forwarding fail. When LDP has not completed exchanging label bindings with an IGP next hop, traffic is discarded if the head end of the LSP forwards traffic because the LSP is assumed to be in place.

There are various reasons that the LSP fails to come up, as follows:

- Configuration errors and implementation issues.
- When an LDP hello adjacency or an LDP session with a peer is lost due to some error while the IGP still points to that peer. IP forwarding of traffic continues on the IGP link associated with the LDP peer rather than being shifted to another IGP link with which LDP is synchronized.
- When a new IGP link comes up, causing the next hop to a certain destination to change in the IGP's shortest-path-first (SPF) calculations. Although the IGP might be up on the new link, LDP might not have completed label exchange for all the routes. This condition might be transient or due to a misconfiguration.

LDP-IGP synchronization discourages a link from being used while the LDP sessions are not fully established. When LDP is not fully operational on a link, the IGP advertises a maximum cost for the link, thus preventing traffic from flowing through it. The IGP does not advertise the original cost or metric for the link until either LDP label exchange has been completed with the peer on the link or a configured amount of time has passed (the holddown period). By default, LDP waits for 10 seconds after establishment of LDP session on the link before notifying IGPs to ensure label exchange has been completed. This is configurable using the [igp-synchronization](#) statement at the `[edit protocols ldp]` hierarchy level.

When synchronization is configured, LDP notifies the IGP to advertise the maximum cost for the link when one of the following triggering events takes place:

- The LDP hello adjacency goes down.
- The LDP session goes down.



- LDP is not configured on an interface.

If the holdown timer has been configured, the timer starts when the triggering event takes place. When the timer expires, LDP notifies the IGP to resume advertising the original cost.

If the holdown timer has not been configured, the IGP waits (endlessly) until LDP sessions are up and has passed to ensure label exchanges have completed. Only after that takes place does LDP notify the IGP to bring down the cost on the interface.

LDP-IGP synchronization is supported only for directly connected peers and links with the platform label space.

### **Synchronization Behavior During Graceful Restart**

LDP-IGP synchronization does not take place while the IGP is in the process of a graceful restart. When the graceful restart completes, links for which synchronization has been configured are advertised with maximum metrics in either of the following cases:

- LDP is not yet operational on the link and no holdown timer has been configured.
- The configured holdown timer has not expired.

During LDP graceful restart, no synchronization operations are done. If the LDP graceful restart is terminated, LDP notifies the IGP to advertise the links with the maximum metric.

### **Synchronization Behavior on LAN Interfaces**

LDP-IGP synchronization does not take place on LAN interfaces unless the IGP has a point-to-point connection over the LAN configured on the interface. The reason for this is that multiple LDP peers might be connected on such an interface unless a point-to-point connection to a single peer has been configured. Because synchronization raises the cost on the interface high enough to prevent traffic from being forwarded to that link, if multiple peers are connected, the cost is raised on all the peers even though LDP might be unsynchronized with only one of the peers. Consequently, traffic is diverted away from all the peers, an undesirable situation.

### **Synchronization Behavior on IGP Passive Interfaces**

On IGP passive interfaces, the link cost is not raised when LDP-IGP synchronization is configured and a triggering event occurs.

## Synchronization and TE Metrics

When traffic engineering is configured for an IGP, LDP-IGP synchronization does not affect the traffic engineering metric advertised for the link, regardless of whether the traffic-engineering (TE) metric is explicitly configured or the default value.

### RELATED DOCUMENTATION

[Example: Configuring Synchronization Between IS-IS and LDP | 452](#)

## Example: Configuring Synchronization Between IS-IS and LDP

### IN THIS SECTION

- [Requirements | 452](#)
- [Overview | 452](#)
- [Configuration | 454](#)
- [Verification | 456](#)

This example shows how to enable synchronization between IS-IS and LDP.

### Requirements

Before you begin, configure IS-IS and LDP. For an example, see *Example: Configuring a Layer 3 VPN with Route Reflection and AS Override*.

### Overview

LDP distributes labels in non-traffic-engineered applications. Labels are distributed along the best path determined by IS-IS. If the synchronization between LDP and IS-IS is lost, the label-switched path (LSP) goes down. Therefore, LDP and IS-IS synchronization is beneficial. When LDP synchronization is configured and when LDP is not fully operational on a given link (a session is not established and labels are not exchanged), IS-IS advertises the link with the maximum cost metric. The link is not preferred but remains in the network topology.

LDP synchronization is supported only on point-to-point interfaces and LAN interfaces configured as point-to-point interfaces under IS-IS. LDP synchronization is not supported during graceful restart.

To advertise the maximum cost metric until LDP is operational for LDP synchronization, include the `ldp-synchronization` statement:

```
ldp-synchronization {
  disable;
  hold-time seconds;
}
```

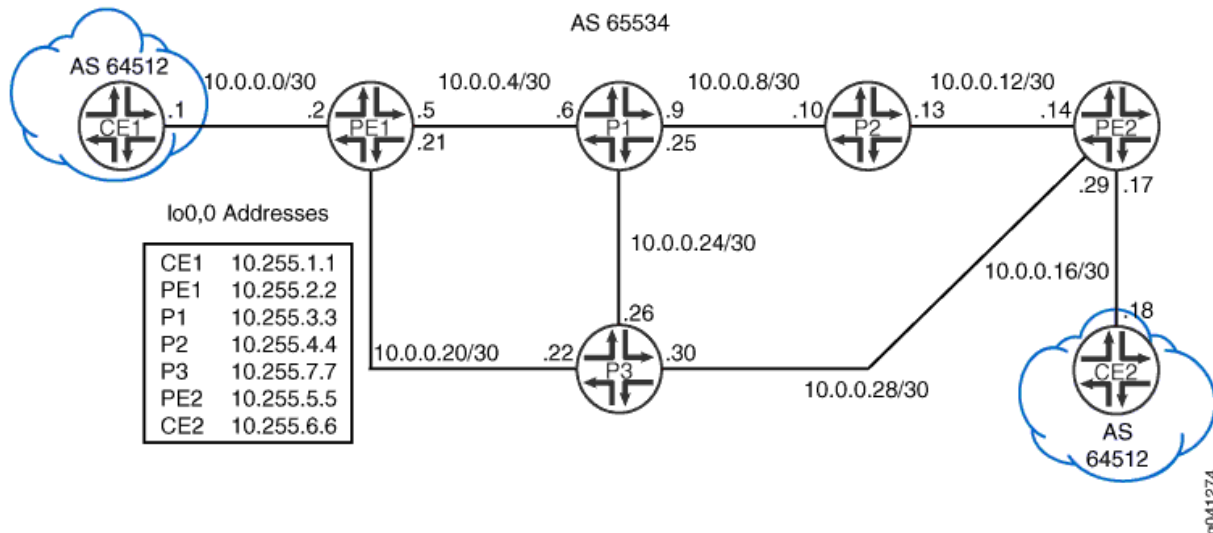
To disable synchronization, include the `disable` statement. To configure the time period to advertise the maximum cost metric for a link that is not fully operational, include the `hold-time` statement.



**NOTE:** When an interface has been in the holddown state for more than 3 minutes, a system log message with a warning level is sent. This message appears in both the messages file and the trace file.

Figure 36 on page 453 shows the topology used in this example.

Figure 36: IS-IS and LDP Synchronization Topology



This example describes the steps on Device P1.

## Configuration

### IN THIS SECTION

- [Procedure](#) | 454

### Procedure

#### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device P1

```
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols isis interface all
set protocols isis interface all ldp-synchronization
set protocols isis interface all point-to-point
set protocols isis interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
```



**NOTE:** We keep the configuration short to better focus on the LDP synchronization feature by using the `interface all` method of enabling protocols on our interfaces. Its a best practice in production networks to explicitly list each interface under each protocol to avoid inadvertently enabling a protocol on an unintended interface.

When using the `interface all` method its a good practice to be consistent in its usage for all protocols, as we show here. In this case it helps ensure you don't inadvertently omit protocol support on an interface that requires the protocol for proper operation.

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure synchronization between IS-IS and LDP:

1. Enable MPLS on the interfaces

```
[edit protocols mpls]
user@P1# set interface all
user@P1# set interface fxp0.0 disable
```

2. Enable IS-IS on the interfaces.

```
[edit protocols isis]
user@P1# set interface all
user@P1# set interface fxp0.0 disable
```

3. Enable LDP on the interfaces.

```
[edit protocols ldp]
user@P1# set interface all
user@P1# set interface fxp0.0 disable
```

4. Enable LDP synchronization on the IS-IS interfaces.

```
[edit protocols isis interface all]
user@P1# set ldp-synchronization
```

5. Configure the IS-IS interfaces to behave like point-to-point interfaces.

```
[edit protocols isis interface all]
user@P1# set point-to-point
```

## Results

From configuration mode, confirm your configuration by entering the `show protocols` command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@P1# show protocols
mpls {
  interface all;
  interface fxp0.0 {
    disable;
  }
}
isis {
  interface all {
    ldp-synchronization;
    point-to-point;
  }
  interface fxp0.0 {
    disable;
  }
}
ldp {
  interface all;
  interface fxp0.0 {
    disable;
  }
}
```

If you are done configuring the device, enter `commit` from configuration mode. Repeat the configuration on Device R2.

## Verification

### IN THIS SECTION

- [Verifying LDP Synchronization | 457](#)

Confirm that the configuration is working properly.

## Verifying LDP Synchronization

### Purpose

Check LDP synchronization setting on the IS-IS interfaces.

### Action

From operational mode, enter the show isis interface extensive command.

```
user@P1> show isis interface extensive
IS-IS interface database:
lo0.0
  Index: 113, State: 0x6, Circuit id: 0x1, Circuit type: 0
  LSP interval: 100 ms, CSNP interval: disabled, Loose Hello padding
  Adjacency advertisement: Advertise
  Level 1
    Adjacencies: 0, Priority: 64, Metric: 0
    Passive
  Level 2
    Adjacencies: 0, Priority: 64, Metric: 0
    Passive
ge-1/2/0.0
  Index: 116, State: 0x6, Circuit id: 0x1, Circuit type: 2
  LSP interval: 100 ms, CSNP interval: 15 s, Loose Hello padding
  Adjacency advertisement: Advertise
  LDP sync state: in sync, for: 17:22:06, reason: LDP up during config
  config holdtime: infinity
  Level 2
    Adjacencies: 1, Priority: 64, Metric: 10
    Hello Interval: 9.000 s, Hold Time: 27 s
ge-1/2/1.0
  Index: 114, State: 0x6, Circuit id: 0x1, Circuit type: 2
  LSP interval: 100 ms, CSNP interval: 15 s, Loose Hello padding
  Adjacency advertisement: Advertise
  LDP sync state: in sync, for: 17:22:06, reason: LDP up during config
  config holdtime: infinity
  Level 2
    Adjacencies: 1, Priority: 64, Metric: 10
    Hello Interval: 9.000 s, Hold Time: 27 s
ge-1/2/2.0
  Index: 115, State: 0x6, Circuit id: 0x1, Circuit type: 2
```

```

LSP interval: 100 ms, CSNP interval: 15 s, Loose Hello padding
Adjacency advertisement: Advertise
LDP sync state: in sync, for: 17:22:06, reason: LDP up during config
config holdtime: infinity
Level 2
  Adjacencies: 1, Priority: 64, Metric: 10
  Hello Interval: 9.000 s, Hold Time: 27 s

```

## Meaning

The output shows that LDP is synchronized with IS-IS.

## RELATED DOCUMENTATION

[Understanding LDP-IGP Synchronization](#) | 449

## Layer 2 Mapping for IS-IS

IS-IS is a Layer 2 protocol that uses the Ethernet logical link control (LLC) encapsulation format for exchanging information. IS-IS Layer 2 mapping ensures that forwarding next-hop resolution is topology-driven rather than traffic-driven, which results in minimal traffic loss while activating an Ethernet link.

Typically, IS-IS installs Layer 3 routes that point to Layer 2 next hops into the forwarding table. Junos OS uses a Layer 3 anchor address notation to standardize the description of a next hop. IS-IS uses Address Resolution Protocol (ARP) to map these IPv4 Layer 3 next-hop anchors to a Layer 2 Media Access Control (MAC) address and installs the Layer 2 MAC addresses in the forwarding table for an Ethernet network. For IPv6 routes, Junos OS uses neighbor discovery to resolve IPv6 Layer 3 next-hop anchors. The Routing Engine installs a Layer 3 prefix along with the set of Layer 3 next-hop anchors for a route in the forwarding table. This method of referencing a Layer 2 next hop using its Layer 3 anchor address in IS-IS networks has the following undesired ramifications:

- When a new route is added to the kernel, its forwarding next hop might not have been resolved yet.
- As next-hop resolution is traffic-driven and always reactive, there is a nonzero traffic loss when you activate an Ethernet link.

Enabling Layer 2 mapping helps to overcome these undesired ramifications in IS-IS networks. IS-IS LAN and point-to-point Hellos supply all relevant Layer 2 and Layer 3 binding address information, which the device at the receiving end can use to populate the ARP or neighbor discovery cache of the kernel even before the route installation time. When Layer 2 mapping is enabled, IS-IS installs ARP or neighbor



discovery next-hop entries into the forwarding table. Because this provides Layer 2 next-hop bindings ahead of time, IS-IS networks do not experience traffic loss while bringing up a link. Each entry gets enqueued as a semi-static ARP or neighbor discovery entry for simplifying garbage collection by a crashed or restarting routing protocol process (rpd). Therefore, each entry gets refreshed periodically.

The advantages of address resolution using IS-IS Hello messages are as follows:

- Forwarding next-hop resolution is topology-driven and not traffic-driven.
- Less Layer 2 resolution on core links because IS-IS already carries this information.
- Better security because IS-IS provides HMAC-MD5 and HMAC-SHA1 digests.



**CAUTION:** The ARP and neighbor discovery methods of address resolution are susceptible to MAC address spoofing attacks.



**NOTE:** Junos OS supports all Ethernet based interface types. However, non-Ethernet based interface types are not supported. Unnumbered IPv4 and IPv6 addresses are not supported as currently IS-IS does not have the capability to generate the IP address neighbor TLVs #132 and #232 from the loopback interface and advertise them on the unnumbered interface.

## RELATED DOCUMENTATION

*layer2-map*

[Example: Configuring Layer 2 Mapping for IS-IS | 459](#)

## Example: Configuring Layer 2 Mapping for IS-IS

### IN THIS SECTION

- [Requirements | 460](#)
- [Overview | 460](#)
- [Configuration | 461](#)
- [Verification | 467](#)

This example shows how to configure Layer 2 mapping for IS-IS, that is, mapping a Layer 2 MAC address to the IPv4 address of the forwarding next hop. Layer 2 mapping minimizes traffic loss, provides better security, and reduces Layer 2 resolution processing on core links while activating an Ethernet link.

## Requirements

This example uses the following hardware and software components:

- Two routers
- Junos OS running on all the devices

## Overview

### IN THIS SECTION

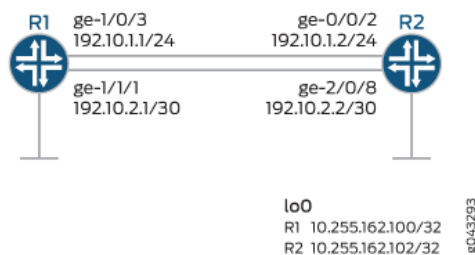
- [Topology](#) | 460

Layer 2 mapping ensures that the forwarding next-hop resolution is topology-driven rather than traffic-driven. IS-IS LAN and point-to-point Hellos supply all relevant Layer 2 and Layer 3 binding address information for address resolution. The device at the receiving end can use the information to populate the ARP or neighbor discovery cache of the kernel even before the route installation time. When Layer 2 mapping is enabled, IS-IS installs ARP or neighbor discovery next-hop entries into the forwarding table. Because this provides Layer 2 next-hop bindings ahead of time, IS-IS networks do not experience traffic loss while bringing up a link.

## Topology

In [Figure 37 on page 461](#), Router R1 is connected to Router R2. Layer 2 mapping is enabled on Router R1. Router R2 receives the Layer 2 information from Router R1 and updates the forwarding table.

Figure 37: Configuring Layer 2 Mapping for IS-IS



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 461](#)
- [Procedure | 463](#)
- [Results | 465](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter `commit` from configuration mode.

#### Router R1

```
set interfaces ge-1/0/3 description R0->R1_1
set interfaces ge-1/0/3 unit 0 family inet address 192.10.1.1/24
set interfaces ge-1/0/3 unit 0 family iso
set interfaces ge-1/0/3 unit 0 family inet6 address 0000:0000:0000:0000:192:10:1:1/120
set interfaces ge-1/0/3 unit 0 family mpls
set interfaces ge-1/0/7 description R0->RT0
set interfaces ge-1/0/7 unit 0 family inet address 193.1.1.1/30
set interfaces ge-1/0/7 unit 0 family iso
set interfaces ge-1/0/7 unit 0 family inet6 address 0000:0000:0000:0000:193:1:1:1/120
set interfaces ge-1/0/7 unit 0 family mpls
set interfaces ge-1/1/1 description R0->R1_2
set interfaces ge-1/1/1 unit 0 family inet address 192.10.2.1/30
```

```

set interfaces ge-1/1/1 unit 0 family iso
set interfaces ge-1/1/1 unit 0 family inet6 address 0000:0000:0000:0000:192:10:2:1/120
set interfaces ge-1/1/1 unit 0 family mpls
set lo0 unit 0 family inet address 10.255.162.100/32
set routing-options router-id 10.255.162.100
set protocols rsvp interface all
set protocols rsvp interface lo0.0
set protocols rsvp interface fxp0.0 disable
set protocols mpls interface all
set protocols mpls interface lo0.0
set protocols mpls interface fxp0.0 disable
set protocols isis layer2-map
set protocols isis interface ge-1/0/3.0 level 2 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set protocols ldp interface lo0.0

```

## Router R2

```

set interfaces ge-0/0/2 description R0->R1_1
set interfaces ge-0/0/2 unit 0 family inet address 192.10.1.2/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/2 unit 0 family inet6 address 0000:0000:0000:0000:192:10:1:2/120
set interfaces ge-0/0/2 unit 0 family mpls
set interfaces ge-2/0/3 description R1->RT0
set interfaces ge-2/0/3 unit 0 family inet address 193.2.1.1/30
set interfaces ge-2/0/3 unit 0 family iso
set interfaces ge-2/0/3 unit 0 family inet6 address 0000:0000:0000:0000:193:2:1:1/120
set interfaces ge-2/0/3 unit 0 family mpls
set interfaces ge-2/0/8 description R0->R1_2
set interfaces ge-2/0/8 unit 0 family inet address 192.10.2.2/30
set interfaces ge-2/0/8 unit 0 family iso
set interfaces ge-2/0/8 unit 0 family inet6 address 0000:0000:0000:0000:192:10:2:2/120
set interfaces ge-2/0/8 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.255.162.102/32
set routing-options router-id 10.255.162.109
set protocols rsvp interface all
set protocols rsvp interface lo0.0
set protocols rsvp interface fxp0.0 disable
set protocols mpls interface all
set protocols mpls interface lo0.0
set protocols mpls interface fxp0.0 disable

```

```

set protocols isis interface ge-0/0/2.0 level 2 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set protocols ldp interface lo0.0

```

## Procedure

### Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [Junos OS CLI User Guide](#).

To configure Layer 2 mapping on Router R1:



**NOTE:** Repeat this procedure for Router 2 after modifying the appropriate interface names, addresses, and other parameters.

#### 1. Configure the device interfaces.

```

[edit interfaces]
user@R1# set ge-1/0/3 description R0->R1_1
user@R1# set ge-1/0/3 unit 0 family inet address 192.10.1.1/24
user@R1# set ge-1/0/3 unit 0 family iso
user@R1# set ge-1/0/3 unit 0 family inet6 address 0000:0000:0000:0000:192:10:1:1/120
user@R1# set ge-1/0/3 unit 0 family mpls
user@R1# set ge-1/0/7 description R0->RT0
user@R1# set ge-1/0/7 unit 0 family inet address 193.1.1.1/30
user@R1# set ge-1/0/7 unit 0 family iso
user@R1# set ge-1/0/7 unit 0 family inet6 address 0000:0000:0000:0000:193:1:1:1/120
user@R1# set ge-1/0/7 unit 0 family mpls
user@R1# set ge-1/1/1 description R0->R1_2
user@R1# set ge-1/1/1 unit 0 family inet address 192.10.2.1/30
user@R1# set ge-1/1/1 unit 0 family iso
user@R1# set ge-1/1/1 unit 0 family inet6 address 0000:0000:0000:0000:192:10:2:1/120
user@R1# set ge-1/1/1 unit 0 family mpls

```

2. Configure the loopback interface.

```
[edit interfaces]
user@R1# set lo0 unit 0 family inet address 10.255.162.100/32
```

3. Configure the router id.

```
[edit routing-options]
user@R1# set router-id 10.255.162.100
```

4. Configure RSVP, MPLS, and LDP on all interfaces excluding the management interface.

```
[edit protocols]
user@R1# set rsvp interface all
user@R1# set rsvp interface lo0.0
user@R1# set rsvp interface fxp0.0 disable
user@R1# set mpls interface all
user@R1# set mpls interface lo0.0
user@R1# set mpls interface fxp0.0 disable
user@R1# set ldp interface all
user@R1# set ldp interface fxp0.0 disable
user@R1# set ldp interface lo0.0
```

5. Enable Layer 2 mapping.

```
[edit protocols]
user@R1# set isis layer2-map
```

6. Disable level 2 IS-IS on interface ge-1/0/3.0.

```
[edit protocols]
user@R1# set isis interface ge-1/0/3.0 level 2 disable
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@R1# show interfaces
ge-1/0/3 {
  description 0->R1_1;
  unit 0 {
    family inet {
      address 192.10.1.1/24;
    }
    family iso;
    family inet6 {
      address 0000:0000:0000:0000:192:10:1:1/120;
    }
    family mpls;
  }
}
ge-1/0/7 {
  description R0->RT0;
  unit 0 {
    family inet {
      address 193.1.1.1/30;
    }
    family iso;
    family inet6 {
      address 0000:0000:0000:0000:193:1:1:1/120;
    }
    family mpls;
  }
}
ge-1/1/1 {
  description R0->R1_2;
  unit 0 {
    family inet {
      address 192.10.2.1/30;
    }
    family iso;
    family inet6 {
```

```

        address 0000:0000:0000:0000:192:10:2:1/120;
    }
    family mpls;
}
lo0 {
    unit 0 {
        family inet {
            address 10.255.162.100/32;
        }
    }
}

```

```

[edit]
user@R1# show protocols
rsvp {
    interface all;
    interface lo0.0;
    interface fxp0.0 {
        disable;
    }
}
mpls {
    interface all;
    interface lo0.0;
    interface fxp0.0 {
        disable;
    }
}
isis {
    layer2-map;
}
ldp {
    interface all;
    interface fxp0.0 {
        disable;
    }
}

```



```
interface lo0.0;
}
```

```
[edit]
user@R1# show routing-options
router-id 10.255.162.100;
```

If you are done configuring the device, commit the configuration.

```
user@R1# commit
```

## Verification

### IN THIS SECTION

- [Verifying IS-IS Adjacencies | 467](#)
- [Verifying That Layer 2 Mapping Is Enabled | 468](#)
- [Verifying That the Layer 2 Address Is Mapped | 468](#)

Confirm that the configuration is working properly.

### Verifying IS-IS Adjacencies

#### Purpose

Verify that the expected adjacencies have formed between Router R1 and Router R2.

#### Action

From operational mode, run the **show isis adjacency** command on Router R1.

```
user@R1> show isis adjacency
Interface          System      L State      Hold (secs) SNPA
ge-1/0/3.0         R2          1 Up          8           88:e0:f3:5e:e8:2
```

## Meaning

The interface ge-1/0/3.0 on Router R1 has established adjacency with Router R2.

## Verifying That Layer 2 Mapping Is Enabled

### Purpose

Verify that Layer 2 mapping is enabled on Router R1.

### Action

From operational mode, run the **show isis interface detail** command on Router R1.

```
user@R1> show isis interface detail
IS-IS interface database:
ge-1/0/3.0
  Index: 196612, State: 0x6, Circuit id: 0x1, Circuit type: 1
  LSP interval: 100 ms, CSNP interval: 10 s
  Adjacency advertisement: Advertise, Layer2-map: Enabled
  Level Adjacencies Priority Metric Hello (s) Hold (s) Designated Router
    1           1       64    10    9.000      27 R2.02 (not us)
    2           0       64    10    Disabled
```

## Meaning

The output confirms that Layer 2 mapping is enabled on Router R1.

## Verifying That the Layer 2 Address Is Mapped

### Purpose

Display Layer 3 next hop and the mapped data link address in the kernel for the routing instances.

### Action

From operational mode, run the **show isis layer2-map** command on Router R1.

```
user@R1> show isis layer2-map
Layer2 mapping database for instance master
```

IP Address	Interface	SNPA	Refresh	State
192.10.1.2	ge-1/0/3.0	88:e0:f3:5e:e8:2	00:11:54	
fe80::8ae0:f3ff:fe5e:e802	ge-1/0/3.0	88:e0:f3:5e:e8:2	00:04:02	
IPv4 records: 1				
IPv6 records: 1				

## Meaning

The Layer 2 MAC address of the next hop is mapped to the IP address of interface ge-1/0/3.0 in the kernel.

## RELATED DOCUMENTATION

*layer2-map*

[Layer 2 Mapping for IS-IS | 458](#)

*show isis layer2-map*

## Overview of Segment Routing Statistics

Traffic statistics in a segment routing network can be recorded in an OpenConfig compliant format for the Layer 3 interfaces. The statistics is recorded for the Source Packet Routing in Networking (SPRING) traffic only, excluding RSVP and LDP-signaled traffic, and the family MPLS statistics per interface is accounted for separately. The SR statistics also includes SPRING traffic statistics per link aggregation group (LAG) member, and per segment identifier (SID). To enable recording of segment routing statistics, include `sensor-based-stats` statement at the `[edit protocol isis source-packet-routing]` hierarchy level.

Earlier, sensors were available for collecting segment routing statistics for MPLS transit traffic only, which is MPLS-to-MPLS in nature. On MX Series routers with MPC and MIC interfaces and PTX Series routers, additional sensors are introduced to collect segment routing statistics for MPLS ingress traffic, which is IP-to-MPLS in nature. With this feature, you can enable sensors for label IS-IS segment routing traffic only, and stream the statistics to a gRPC client.

You can enable the segment routing statistics for MPLS ingress traffic using the `egress` option under the `per-sid` configuration statement. The resource name for the `per-sid egress` functionality is:

```
/junos/services/segment-routing/sid/egress/usage/
```

You can view the label IS-IS route association with the sensors using the `show isis spring sensor info` command output. This command does not display counter values of the actual sensors.

The segment routing statistics records are exported to a server. You can view segment routing statistics data from the following the OpenConfig paths:

- `/mpls/signalling-protocols/segment-routing/aggregate-sid-counters/aggregate-sid-counter[ip-addr='L-ISIS-10.1.1.1']/state/counters[name='oc-xxx']/out-pkts`
- `/mpls/signalling-protocols/segment-routing/aggregate-sid-counters/aggregate-sid-counter[ip-addr='L-ISIS-10.1.1.1']/state/counters[name='oc-xxx']/out-pkts`



#### NOTE:

- Graceful Routing Engine switchover (GRES) is not supported for segment routing statistics.

Nonstop active routing (NSR) is not supported for label IS-IS. During a Routing Engine switchover, a new sensor is created in the new primary Routing Engine, replacing the sensor created by the previous primary Routing Engine. As a result, at the time of a Routing Engine switchover, the segment routing statistics counter start from zero.

- Graceful restart is not supported for label IS-IS.

In case of graceful restart, the existing sensor is deleted and a new sensor is created during IS-IS initialization. The segment routing statistics counter restarts from zero.

- In-service software upgrade (ISSU) and nonstop software upgrade (NSSU) are not supported. In such cases, the segment routing statistics counter is restarted.
- Zero-statistics segment routing data is suppresses and does not get streamed to the gRPC clients.

## RELATED DOCUMENTATION

[sensor-based-stats](#)

[sensor \(Junos Telemetry Interface\)](#)

[sensor-based-stats \(Junos Telemetry Interface\)](#)

## Overview of Adjacency Segments, Anycast Segments, and Configurable SRGB in SPRING

### IN THIS SECTION

- [Benefits of Anycast Segments, Adjacency Segments, and Configurable SRGB | 472](#)
- [Configurable Segment Routing Global Block | 472](#)
- [Adjacency Segments and Prefix Segments | 472](#)
- [Prefix SID redistribution into OSPF through policy configuration | 474](#)

Segment routing (SR) or Source Packet Routing in Networking (SPRING) is a control-plane architecture that enables an ingress router to steer a packet through a specific set of nodes and links without relying on the intermediate nodes in the network to determine the actual path it should take. SPRING enables automation of a network by using a software-defined network (SDN) controller for traffic steering and traffic engineering in a WAN packet network. To steer packets through the specified set of nodes and links, the ingress router prepends packets with segments that contain an appropriate combination of tunnels. Each segment is associated with an identifier, which is referred to as the segment identifier (SID). An ordered list of segments is encoded as a stack of labels. Every node in the segment routing domain is allocated labels based on the availability of the dynamic label range. A segment routing global block (SRGB) is the range of label values reserved for segment routing.

Configure the SRGB range label for SPRING. These labels are used by SPRING within the IS-IS or OSPF domain. This way the labels advertised in segment routing are more predictable and deterministic across the segment routing domain. To configure the starting index value of the SRGB label block, configure:

- The `start-label start-label-block-value` statement at the `[edit protocols isis source-packet-routing srgb]` hierarchy level for IS-IS.
- The `start-label start-label-block-value` statement at the `[edit protocols ospf source-packet-routing srgb]` hierarchy level for OSPF.

To configure the index range of the SRGB label block, configure:

- The `index-range value` statement at the `[edit protocols isis source-packet-routing srgb]` hierarchy level for IS-IS.
- The `index-range value` statement at the `[edit protocols isis source-packet-routing srgb]` hierarchy level for IS-IS.

Define the SRGB for the IS-IS protocol, and provide prefix anycast segments in addition to node segments to prefixes that are advertised by the IS-IS protocol through policy configuration. Junos OS also extends support to SPRING anycast segments and configurable adjacency segment indexes for the IS-IS protocol.

## Benefits of Anycast Segments, Adjacency Segments, and Configurable SRGB

- With the support for anycast prefix segments on Junos OS, you can configure multiple routers to advertise the same prefix with the same SID, which facilitates load balancing.
- Configuring the adjacency hold time helps retain segments for a specified period of time after a link flaps and ensures faster convergence after a link fails.
- Configuring the SRGB label range ensures that the labels are more predictable across segment routing domain.

## Configurable Segment Routing Global Block

A segment is encoded as an MPLS label. An ordered list of segments is encoded as a stack of labels. Every node in the segment routing domain is allocated labels by the node label manager based on the index range configured for source packet routing. These labels are allocated to the node segment based on the availability of the dynamic label range managed by node label manager. An SRGB is the range of label values used in segment routing.

Junos OS supports SRGB under `mpls label-range`. You can configure an available SRGB label range for the IS-IS, OSPF, and BGP protocols so that the labels are predictable across segment routing domains. Ensure that the configured SRGB labels are not used by any other application.

To configure SRGB under the global MPLS label range, execute the following command: `set protocols mpls-label-range srgb range-start end range-end`. You don't need separate label ranges for each protocol because Junos OS centralizes SRGB under MPLS. This enables IS-IS, OSPF, and BGP share one predictable pool, conserving label space

## Adjacency Segments and Prefix Segments

A node steers a packet to its destination through an ordered list of instructions, called segments. Essentially, segment routing engages interior gateway protocols (IGPs) such as IS-IS and OSPF to advertise two types of network segments:

- Adjacency segments—A strict forwarded single-hop tunnel that carries packets over a specific link between two nodes, irrespective of the link cost.
- Prefix segments—A multihop tunnel that uses equal cost multi-hop aware shortest path links to reach a prefix. The prefix SID supports both IPv4 and IPv6 prefixes. A node segment is a special case of

prefix segment, where every node computes shortest path to the node segment and programs in the forwarding plane. An anycast segment is also a type of prefix segment that identifies a set of routers to advertise the same prefix with the same SID value.

You can redistribute Segment Routing (SR) prefix-Segment Identifiers (SIDs) across OSPF IGP instances using route policy without explicitly specifying a prefix-segment index. This feature standardizes SR labels across instances and improves operational efficiency. Configure a policy with the `from prefix-segment` statement to match routes carrying prefix-segment information. In the `then` clause, use `prefix-segment redistribute` to inherit segment information from the matched route. We also support stitching `mpls.0` routes to enable interoperability between different IGP instances.

## Configurable Adjacency Segment Hold Time

The IS-IS protocol creates adjacency segments per adjacency, level, and address family (one each for IPv4 and IPv6). An MPLS label is allocated for each adjacency segment that gets created. These labels are allocated after the adjacency status of the segment changes to the up state. Configure a hold time to ensure that IS-IS does not release the segments immediately after a link flaps or goes down, but retains them for the configured hold time duration. The default hold time for adjacency segments in IS-IS protocol is 300 seconds.

The OSPF protocol creates adjacency segments per adjacency. To ensure adjacency segments are retained during adjacency or link flaps, the adjacency segments are not released immediately during the link down. The default hold time for adjacency segments in OSPF protocol is 180 seconds.

## Prefix Segment Index

Currently, Junos OS enables you to configure a SPRING node SID for IPv4 and IPv6 address families for each routing instance. This node SID is attached to an IPv4 and IPv6 router ID if the router ID is configured on the loopback interface. Otherwise, the lowest IP address assigned to the loopback interface is chosen as the node SID. Configuring a node SID through policy allows you to choose the loopback address that gets the node SID. If the node SID configuration exists and a policy is defined for node SID selection for the same prefix, then the policy configuration takes precedence.

Designate prefix segment indexes to prefix SIDs, both anycast and node SIDs, that are advertised in IS-IS through policy configuration. Remote routers use the SRGB and the index to derive labels for a specific prefix. After the prefix segment indexes are provisioned, the devices running Junos OS advertise them in one or more of the following IS-IS TLV types by using a new Prefix-SID Sub-TLV (type 3):

- IP Prefix TLV (type 135)
- MT IP Prefix TLV (type 235)
- IPV6 Prefix Reachability TLV (type 236)
- MT IPV6 Prefix Reachability TLV (type 237)

You can similarly designate prefix segment indexes to prefix SIDs, both anycast and node SIDs, that are advertised in OSPF through policy configuration. Remote routers use this index to consolidate prefixes into respective SRGBs and to derive the segment identifier and forward the traffic destined for a specific prefix.

Anycast Segments

An IGP anycast segment is an IGP prefix segment that identifies a set of routers. An anycast segment enforces forwarding based on the equal-cost multipath-aware shortest-path toward the closest node of the anycast set. Within an anycast group, all the routers advertise the same prefix with the same SID value, which facilitates load balancing.

Prefix SID redistribution into OSPF through policy configuration

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
25.4R1	Starting in Junos OS and Junos OS Evolved Release 25.4R1, you can redistribute Segment Routing (SR) prefix-Segment Identifiers (SIDs) across OSPF IGP instances using route policy without explicitly specifying a prefix-segment index.

RELATED DOCUMENTATION

<i>Configuring Anycast and Prefix segments in SPRING for IS-IS Protocol</i>
<i>Configuring Segment Routing Global Blocks Label Ranges in SPRING for IS-IS Protocol</i>
<i>Example: Configuring Segment Routing Global Blocks in SPRING for IS-IS to Increase Network Speed</i>
<i>prefix-segment</i>
<i>srgb</i>
<i>traffic-engineering</i>



## Example: Configuring SRGB in Segment Routing for IS-IS

### IN THIS SECTION

- Requirements | 475
- Overview | 476
- Configuration | 476
- Verification | 481

This example shows how to define the segment routing label block (SRGB) label range for segment packet routing in networking (SPRING) or segment routing (SR) for the IS-IS protocol. This configuration ensures that the labels are more predictable across the segment routing domain with a beneficial impact on network speed.



**NOTE:** Our content testing team has validated and updated this example.

### Requirements

This example uses the following hardware and software components:

- Two MX Series routers
- Junos OS Release 17.2 or later running on all devices
  - Updated and revalidated using vMX on Junos OS Release 21.1R1.



**NOTE:** Are you interested in getting hands-on experience on this feature?

Visit Juniper vLabs to reserve your pre-configured [vLab Sandbox: Segment Routing - Basic](#) and try it out for free!

Before you configure the SRGB label range for segment routing in the IS-IS domain, ensure that you've configured the routing and signaling protocols.

## Overview

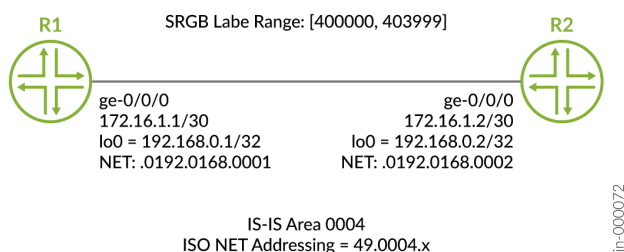
### IN THIS SECTION

- [Topology | 476](#)

Currently, Junos OS allows you to configure only node segment indices. The value of the start label depends on the dynamic label available in the system. Because there is no predictability of the dynamic label range being allocated to the SRGB, Junos OS allows you to configure the SRGB label range used by segment routing. The labels in the SRGB range are used for segment routing in the IS-IS domain. This means the labels advertised are more predictable and deterministic across the segment routing domain.

### Topology

Figure 1 shows SRGB configured on router R1 and router R2.



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 477](#)
- [Configuring Device R1 | 478](#)
- [Results | 480](#)

## CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter `commit` from configuration mode.

### R1

```
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 172.16.1.1/30
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family inet6 address 2001:db8:1:1::1/128
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0004.0192.0168.0001.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:10:10::1/128
set protocols isis interface ge-0/0/0.0
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 4000
set protocols isis source-packet-routing node-segment ipv4-index 2001
set protocols isis source-packet-routing node-segment ipv6-index 3001
set protocols isis level 1 disable
set protocols mpls interface ge-0/0/0.0
```

### R2

```
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 172.16.1.2/30
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family inet6 address 2001:db8:1:1::2/64
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0004.0192.0168.0002.00
set interfaces lo0 unit 0 family inet6 address 2001:db8:20:20::1/128
set protocols isis interface ge-0/0/0.0
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 4000
set protocols isis source-packet-routing node-segment ipv4-index 2002
```

```
set protocols isis source-packet-routing node-segment ipv6-index 3002
set protocols isis level 1 disable
set protocols mpls interface ge-0/0/0.0
```

## Configuring Device R1

### Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [Junos OS CLI User Guide](#).

To configure device R1:



**NOTE:** Repeat this procedure for device R2 after modifying the appropriate interface names, addresses, and other parameters.

1. Configure enhanced-ip mode on the MX Series because the SRGB functionality is supported on routers with MPCs and MIC interfaces only. A system reboot is required after you commit this configuration.

```
[edit chassis]
user@R1# set network-services enhanced-ip
```

2. Configure the interfaces.

```
[edit interfaces]
user@R1# set ge-0/0/0 unit 0 family inet address 172.16.1.1/30
user@R1# set ge-0/0/0 unit 0 family iso
user@R1# set ge-0/0/0 unit 0 family inet6 address 2001:db8:1:1::1/128
user@R1# set ge-0/0/0 unit 0 family mpls
user@R1# set lo0 unit 0 family inet address 192.168.0.1/32
user@R1# set lo0 unit 0 family iso address 49.0004.0192.0168.0001.00
user@R1# set lo0 unit 0 family inet6 address 2001:db8:10:10::1/128
```

3. Configure the MPLS protocol on the interface. For segment routing to work, you can configure any of the statements under the `[edit protocols mpls]` hierarchy.

```
[edit protocols]
user@R1# set mpls interface ge-0/0/0.0
```

4. Configure the start label and index range of SRGB.



**NOTE:**

- Ensure that the MPLS label for a binding segment ID (SID) is the sum of the SRGB start label and SID index value. In addition, SID index value must be less than or equal to the index-range value specified in the configuration.
- Junos does not check whether the SID index is within the SRGB's range when the SID index is assigned through an IS-IS export policy. If you configure an index that is out of range of the configured SRGB, you won't see any error message in the logs or while committing the configuration. Junos OS shows a commit error only when you configure the SID under the `[edit protocols isis source-packet-routing]` hierarchy level.

```
[edit protocols]
user@R1# set isis source-packet-routing srgb start-label 400000
user@R1# set isis source-packet-routing srgb index-range 4000
```

5. Configure the IPv4 index value of the node segment.

```
[edit protocols]
user@R1# set isis source-packet-routing node-segment ipv4-index 2001
```

6. Configure the IPv6 index value of the node segment.

```
[edit protocols]
user@R1# set isis source-packet-routing node-segment ipv6-index 3001
```

7. Disable level 1, configure the IS-IS protocol on the interface, and configure loopback interface lo0.0 as passive..

```
[edit protocols]
user@R1# set isis level 1 disable
user@R1# set isis interface ge-0/0/0.0
user@R1# set isis interface lo0.0 passive
```

## Results

From configuration mode, confirm your configuration by entering the **show chassis**, **show interfaces**, and **show protocols** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show chassis
network-services enhanced-ip;
```

```
user@R1# show interfaces
ge-0/0/0 {
  unit 0 {
    family inet {
      address 172.16.1.1/30;
    }
    family iso;
    family inet6 {
      address 2001:db8:1:1::1/128;
    }
    family mpls;
  }
}
lo0 {
  unit 0 {
    family inet {
      address 192.168.0.1/32;
    }
    family iso {
      address 49.0004.0192.0168.0001.00;
    }
    family inet6 {
```

```

        address 2001:db8:10:10::1/128;
    }
}

```

```

user@R1# show protocols
isis {
    interface ge-0/0/0.0;
    interface lo0.0 {
        passive;
    }
    source-packet-routing {
        srgb start-label 400000 index-range 4000;
        node-segment {
            ipv4-index 2001;
            ipv6-index 3001;
        }
    }
    level 1 disable;
}
mpls {
    interface ge-0/0/0.0;
}

```

## Verification

### SUMMARY

Confirm that the configuration is working properly.

### IN THIS SECTION

- [Verifying the Configurable SRGB | 481](#)

## Verifying the Configurable SRGB

### Purpose

Verify the configurable SRGB label range in the IS-IS overview information.

## Action

From operational mode, run the `show isis overview` command to display the IS-IS overview information.

```

user@R1> show isis overview
Instance: master
  Router ID: 128.53.50.230
  IPv6 Router ID: abcd::128:53:50:230
  Hostname: R1
  Sysid: 1280.5305.0230
  Areaid: 47.0005.80ff.f800.0000.0108.0001
  Adjacency holddown: enabled
  Maximum Areas: 3
  LSP life time: 1200
  Attached bit evaluation: enabled
  SPF delay: 200 msec, SPF holddown: 5000 msec, SPF rapid runs: 3
  IPv4 is enabled, IPv6 is enabled, SPRING based MPLS is enabled
  Traffic engineering: enabled
  Traffic engineering v6: disabled
  Restart: Disabled
    Helper mode: Enabled
  Layer2-map: Disabled
  Source Packet Routing (SPRING): Enabled
    SRGB Config Range :
      SRGB Start-Label : 400000, SRGB Index-Range : 4000
    SRGB Block Allocation: Success
      SRGB Start Index : 400000, SRGB Size : 4000, Label-Range: [ 400000, 403999 ]
  Node Segments: Enabled
    Ipv4 Index : 2001, Ipv6 Index : 3001
  SRv6: Disabled
  Post Convergence Backup: Disabled
  Level 1
    Internal route preference: 15
    External route preference: 160
    Prefix export count: 0
    Wide metrics are enabled, Narrow metrics are enabled
    Source Packet Routing is enabled
  Level 2
    Internal route preference: 18
    External route preference: 165
    Prefix export count: 0

```



Wide metrics are enabled, Narrow metrics are enabled  
Source Packet Routing is enabled

## Meaning

The output displays the configured SRGB start label and the SRGB index range. The end of the SRGB label range is the summation of the start label value and the index range. All devices in the segment routing domain must have the same SRGB range values.

## RELATED DOCUMENTATION

[Overview of Adjacency Segments, Anycast Segments, and Configurable SRGB in SPRING | 471](#)

[Configure Segment Routing Global Blocks Label Ranges in SPRING for IS-IS Protocol | 504](#)

*source-packet-routing*

[vLab Sandbox: Segment Routing - Basic](#)

## Example: Configuring Anycast and Prefix Segments in SPRING for IS-IS

### IN THIS SECTION

- [Requirements | 483](#)
- [Overview | 484](#)
- [Configuration | 485](#)
- [Verification | 498](#)

This example shows how to configure prefix segments, segment-routing global blocks (SRGBs), adjacency segments hold time, and explicit null flag for prefix segments in source packet routing in networking (SPRING) or segment routing (SR). This configuration helps in simplifying the network thereby increasing the speed of the network.

## Requirements

This example uses the following hardware and software components:

- Eight MX Series routers.

- Junos OS Release 17.2 or later running on all devices.

Before you configure prefix segments in SPRING, be sure you configure routing and signaling protocols.

## Overview

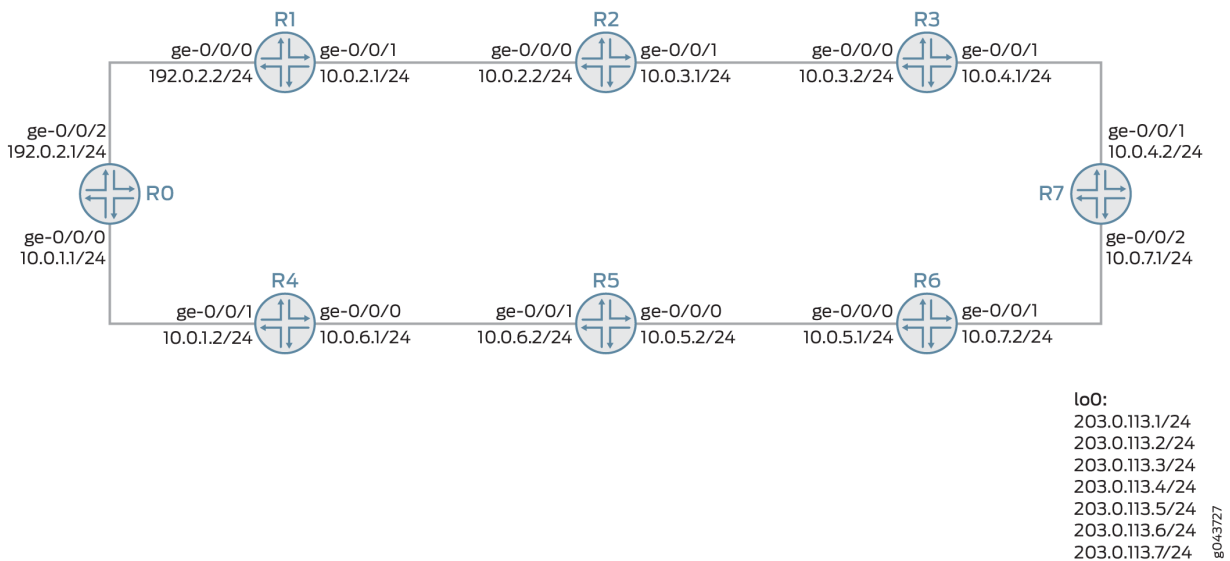
### IN THIS SECTION

- [Topology | 484](#)

You can provide prefix segment identifier (SID) and node SID to prefixes that are advertised in IS-IS by configuring policies. Prefix segment index is the index assigned to a specific prefix. This is used by all other remote routers in the network to index the prefix into respective segment-routing global blocks (SRGBs) to derive the segment identifier and to forward the traffic destined for this prefix. The prefix SID supports both IPv4 and IPv6 prefixes. The IS-IS protocol creates adjacency segments per adjacency, level, and address family (one each for IPv4 and IPv6).

## Topology

Figure 1 shows SRGBs, prefix segments, and adjacency hold time configured in SPRING on routers R0 to R7.



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 485](#)
- [Configuring Router R4 | 493](#)
- [Results | 496](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.



**NOTE:** This topology demonstrates IPv4 prefixes. The same is applicable for IPv6 prefixes.

### R0

```
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 1 family inet address 10.0.1.1/24
set interfaces ge-0/0/0 unit 1 family iso
set interfaces ge-0/0/2 vlan-tagging
set interfaces ge-0/0/2 unit 1 vlan-id 1
set interfaces ge-0/0/2 unit 1 family inet address 192.10.12.1/24
set interfaces ge-0/0/2 unit 1 family iso
set interfaces ge-0/0/2 unit 1 family mpls maximum-labels 5
set interfaces lo0 unit 0 family inet address 203.0.113.1/24
set interfaces lo0 unit 0 family iso address 49.0001.000a.0a0a.0a00
set routing-options autonomous-system 100
set routing-options router-id 203.0.113.1
set routing-options forwarding-table export pplb
set routing-options forwarding-table chained-composite-next-hop ingress l3vpn
set protocols mpls traffic-engineering
set protocols mpls interface all
set protocols isis export prefix-sid
```

```

set protocols isis backup-spf-options remote-backup-calculation
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis graceful-restart restart-duration 30
set protocols isis source-packet-routing adjacency-segment hold-time 240000
set protocols isis source-packet-routing srgb start-label 800000
set protocols isis source-packet-routing srgb index-range 40000
set protocols isis source-packet-routing explicit-null

set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 203.0.113.1/24 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1000
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

```

## R1

```

set chassis network-services enhanced-ip
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 1 family inet address 192.0.2.2/24
set interfaces ge-0/0/0 unit 1 family iso
set interfaces ge-0/0/0 unit 1 family mpls
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 unit 1 vlan-id 1
set interfaces ge-0/0/1 unit 1 family inet address 10.0.2.1/24
set interfaces ge-0/0/1 unit 1 family iso
set interfaces ge-0/0/1 unit 1 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.2/24
set interfaces lo0 unit 0 family iso address 49.0001.0001.0101.0100
set routing-options router-id 203.0.113.2
set routing-options forwarding-table export pplb
set protocols mpls traffic-engineering
set protocols mpls interface all
set protocols isis export prefix-sid
set protocols isis backup-spf-options per-prefix-calculation
set protocols isis backup-spf-options remote-backup-calculation
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis traffic-engineering family inet shortcuts

```

```

set protocols isis graceful-restart restart-duration 30
set protocols isis source-packet-routing adjacency-segment hold-time 240000
set protocols isis source-packet-routing srgb start-label 800000
set protocols isis source-packet-routing srgb index-range 40000
set protocols isis source-packet-routing explicit-null

set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis label-switched-path to_r2
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 203.0.113.2/24 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1001
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept
set policy-options policy-statement setpref from protocol isis
set policy-options policy-statement setpref from level 2
set policy-options policy-statement setpref then preference 11
set policy-options policy-statement setpref then local-preference 11
set policy-options policy-statement setpref then accept

```

## R2

```

set chassis network-services enhanced-ip
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 encapsulation flexible-ethernet-services
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 1 family inet address 10.0.2.2/24
set interfaces ge-0/0/0 unit 1 family iso
set interfaces ge-0/0/0 unit 1 family mpls
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 encapsulation flexible-ethernet-services
set interfaces ge-0/0/1 unit 1 vlan-id 1
set interfaces ge-0/0/1 unit 1 family inet address 10.0.3.1/24
set interfaces ge-0/0/1 unit 1 family iso
set interfaces ge-0/0/1 unit 1 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.3/24
set interfaces lo0 unit 0 family iso address 49.0001.0002.0202.0200
set routing-options router-id 203.0.113.3
set routing-options forwarding-table export pplb
set protocols mpls interface all

```

```

set protocols isis export leakl2tol1
set protocols isis export prefix-sid
set protocols isis backup-spf-options remote-backup-calculation
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis source-packet-routing adjacency-segment hold-time 240000
set protocols isis source-packet-routing srgb start-label 800000
set protocols isis source-packet-routing srgb index-range 40000
set protocols isis source-packet-routing explicit-null

set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis label-switched-path to_r1
set policy-options policy-statement leakl2tol1 from protocol isis
set policy-options policy-statement leakl2tol1 from level 2
set policy-options policy-statement leakl2tol1 to protocol isis
set policy-options policy-statement leakl2tol1 to level 1
set policy-options policy-statement leakl2tol1 then accept
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 203.0.113.3/24 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1002
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

```

### R3

```

set chassis network-services enhanced-ip
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 1 family inet address 10-.0.3.2/24
set interfaces ge-0/0/0 unit 1 family iso
set interfaces ge-0/0/0 unit 1 family mpls
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 unit 1 vlan-id 1
set interfaces ge-0/0/1 unit 1 family inet address 10.0.4.1/24
set interfaces ge-0/0/1 unit 1 family iso
set interfaces ge-0/0/1 unit 1 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.4/24
set interfaces lo0 unit 0 family iso address 49.0001.0003.0303.0300
set routing-options router-id 203.0.113.4
set routing-options forwarding-table export pplb

```

```

set protocols mpls interface all
set protocols isis export prefix-sid
set protocols isis backup-spf-options remote-backup-calculation
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis source-packet-routing adjacency-segment hold-time 240000
set protocols isis source-packet-routing srgb start-label 800000
set protocols isis source-packet-routing srgb index-range 40000
set protocols isis source-packet-routing explicit-null

set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 203.0.113.4/24 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1003
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

```

#### R4

```

set chassis network-services enhanced-ip
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 1 family inet address 10.0.6.2/24
set interfaces ge-0/0/0 unit 1 family iso
set interfaces ge-0/0/0 unit 1 family mpls
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 unit 1 vlan-id 1
set interfaces ge-0/0/1 unit 1 family inet address 10.0.1.2/24
set interfaces ge-0/0/1 unit 1 family iso
set interfaces ge-0/0/1 unit 1 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.5/24
set interfaces lo0 unit 0 family iso address 49.0001.0004.0404.0400
set routing-options router-id 203.0.113.5
set routing-options forwarding-table export pplb

set protocols mpls interface all
set protocols isis export prefix-sid
set protocols isis backup-spf-options remote-backup-calculation
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis source-packet-routing adjacency-segment hold-time 240000

```

```

set protocols isis source-packet-routing srgb start-label 800000
set protocols isis source-packet-routing srgb index-range 40000
set protocols isis source-packet-routing explicit-null

set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 203.0.113.5/24 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1004
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

```

## R5

```

set chassis network-services enhanced-ip
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 1 family inet address 10.0.5.2/24
set interfaces ge-0/0/0 unit 1 family iso
set interfaces ge-0/0/0 unit 1 family mpls
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 unit 1 vlan-id 1
set interfaces ge-0/0/1 unit 1 family inet address 10.0.6.2/24
set interfaces ge-0/0/1 unit 1 family iso
set interfaces ge-0/0/1 unit 1 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.6/24
set interfaces lo0 unit 0 family iso address 49.0001.0005.0505.0500
set routing-options router-id 203.0.113.6
set routing-options forwarding-table export pplb
set protocols mpls interface all
set protocols isis export leakl2tol1
set protocols isis export prefix-sid
set protocols isis backup-spf-options remote-backup-calculation
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis source-packet-routing adjacency-segment hold-time 240000
set protocols isis source-packet-routing srgb start-label 800000
set protocols isis source-packet-routing srgb index-range 40000
set protocols isis source-packet-routing explicit-null

```



```

set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set policy-options policy-statement leakl2tol1 from protocol isis
set policy-options policy-statement leakl2tol1 from level 2
set policy-options policy-statement leakl2tol1 to protocol isis
set policy-options policy-statement leakl2tol1 to level 1
set policy-options policy-statement leakl2tol1 then accept
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 203.0.113.6/24 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1005
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

```

## R6

```

set chassis network-services enhanced-ip
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 1 vlan-id 1
set interfaces ge-0/0/0 unit 1 family inet address 10.0.5.1/24
set interfaces ge-0/0/0 unit 1 family iso
set interfaces ge-0/0/0 unit 1 family mpls
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 unit 1 vlan-id 1
set interfaces ge-0/0/1 unit 1 family inet address 10.0.6.2/24
set interfaces ge-0/0/1 unit 1 family iso
set interfaces ge-0/0/1 unit 1 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.7/24
set interfaces lo0 unit 0 family iso address 49.0001.0006.0606.0600
set routing-options router-id 203.0.113.7
set routing-options forwarding-table export pplb
set protocols mpls interface all
set protocols isis export prefix-sid
set protocols isis backup-spf-options remote-backup-calculation
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis source-packet-routing adjacency-segment hold-time 240000
set protocols isis source-packet-routing srgb start-label 800000
set protocols isis source-packet-routing srgb index-range 40000
set protocols isis source-packet-routing explicit-null

```

```

set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 203.0.113.7/24 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1006
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

```

## R7

```

set chassis network-services enhanced-ip
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 unit 1 vlan-id 1
set interfaces ge-0/0/1 unit 1 family inet address 10.0.4.2/24
set interfaces ge-0/0/1 unit 1 family iso
set interfaces ge-0/0/1 unit 1 family mpls
set interfaces ge-0/0/2 vlan-tagging
set interfaces ge-0/0/2 unit 1 vlan-id 1
set interfaces ge-0/0/2 unit 1 family inet address 10.0.7.1/24
set interfaces ge-0/0/2 unit 1 family iso
set interfaces ge-0/0/2 unit 1 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.8/24
set interfaces lo0 unit 0 family iso address 49.0001.0007.0707.0700
set routing-options router-id 203.0.113.8
set routing-options autonomous-system 100
set routing-options forwarding-table export pplb
set protocols mpls explicit-null
set protocols mpls interface all
set protocols isis export prefix-sid
set protocols isis backup-spf-options remote-backup-calculation
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis source-packet-routing adjacency-segment hold-time 240000
set protocols isis source-packet-routing srgb start-label 800000
set protocols isis source-packet-routing srgb index-range 40000
set protocols isis source-packet-routing explicit-null

set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 203.0.113.8/24 exact

```

```

set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1007
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept
set policy-options policy-statement setpref from protocol isis
set policy-options policy-statement setpref from level 2
set policy-options policy-statement setpref then preference 11
set policy-options policy-statement setpref then local-preference 11
set policy-options policy-statement setpref then accept
set policy-options policy-statement stat term 1 from protocol static
set policy-options policy-statement stat term 1 then accept

```

## Configuring Router R4

### Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [Junos OS CLI User Guide](#).

To configure Router R4:



**NOTE:** Repeat this procedure for every router in the SPRING domain, after modifying the appropriate interface names, addresses, and any other parameters for each router.

1. Configure enhanced IP mode on the MX Series router because the SRGB functionality is supported on routers with MPCs and MIC interfaces only. A system reboot is required after you commit this configuration.

```

[edit chassis]
user@R4# set network-services enhanced-ip

```

2. Configure the interfaces.

```

[edit interfaces]
user@R4# set ge-0/0/0 vlan-tagging
user@R4# set ge-0/0/0 unit 1 vlan-id 1
user@R4# set ge-0/0/0 unit 1 family inet address 10.0.6.2/24
user@R4# set ge-0/0/0 unit 1 family iso
user@R4# set ge-0/0/0 unit 1 family mpls
user@R4# set ge-0/0/1 vlan-tagging

```

```

user@R4# set ge-0/0/1 unit 1 vlan-id 1
user@R4# set ge-0/0/1 unit 1 family inet address 10.0.1.2/24
user@R4# set ge-0/0/1 unit 1 family iso
user@R4# set ge-0/0/1 unit 1 family mpls
user@R4# set lo0 unit 0 family inet address 203.0.113.5/24
user@R4# set lo0 unit 0 family iso address 49.0001.0004.0404.0400

```

3. Configure the router ID for a routing option.

```

[edit routing-options]
user@R4# set router-id 203.0.113.5

```

4. Configure the export policy for the forwarding table.

```

[edit routing-options]
user@R4# set forwarding-table export pplb

```

5. Configure the MPLS interface.

```

[edit protocols mpls]
user@R4# set interface all

```

6. Configure the export policy for the IS-IS protocol.

```

[edit protocols isis]
user@R4# set export prefix-sid

```

7. Configure backup shortest-path-first options to calculate remote loop-free alternate (LFA) backup next hops and to use SPRING routed paths for protection for the IS-IS protocol.

```

[edit protocols isis]
user@R4# set backup-spf-options remote-backup-calculation
user@R4# set backup-spf-options use-source-packet-routing

```

8. Configure adjacency segment hold time in SPRING for the IS-IS protocol.

```
[edit protocols isis]
user@R4# set source-packet-routing adjacency-segment hold-time 240000
```

9. Configure the start label and index range for segment routing global blocks (SRGBs) in SPRING for the IS-IS protocol.

```
[edit protocols isis]
user@R4# set source-packet-routing srgb start-label 800000
user@R4# set source-packet-routing srgb index-range 40000
```

10. Configure explicit null in SPRING for the IS-IS protocol.

```
[edit protocols isis]
user@R4# set source-packet-routing explicit-null
```

11. Disable the management interface and configure the loopback address as passive for the IS-IS protocol.

```
[edit protocols isis]
user@R4# set interface fxp0.0 disable
user@R4# set interface lo0.0 passive
```

12. Configure per packet load balancing for the routing policy.

```
[edit policy-options policy-statement pplb]
user@R4# set then load-balance per-packet
```

13. Configure the route filter for the routing policy term.

```
[edit policy-options policy-statement prefix-sid]
user@R4# set term 1 from route-filter 203.0.113.5/24 exact
```

14. Configure the index and node segment of the prefix segment for the routing policy term.

```
[edit policy-options policy-statement prefix-sid]
user@R4# set term 1 then prefix-segment index 1004
user@R4# set term 1 then prefix-segment node-segment
user@R4# set term 1 then accept
```

## Results

From configuration mode, confirm your configuration by entering the **show chassis**, **show interfaces**, **show protocols**, **show policy-options**, and **show routing-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R4# show chassis
network-services enhanced-ip;
```

```
user@R4# show interfaces
ge-0/0/0 {
  vlan-tagging;
  unit 1 {
    vlan-id 1;
    family inet {
      address 10.0.6.2/24;
    }
    family iso;
  }
}
ge-0/0/1 {
  vlan-tagging;
  unit 1{
    vlan-id 1;
    family inet {
      address 10.0.1.2/24;
    }
    family iso;
    family mpls ;
  }
}
lo0 {
```

```

unit 0 {
    family inet {
        address 203.0.113.5/24;
    }
    family iso {
        address 49.0001.0004.0404.0400;
    }
}
}

```

```

user@R4# show protocols
mpls {
    interface all;
}
isis {
    export prefix-sid;
    backup-spf-options {
        remote-backup-calculation;
        use-source-packet-routing;
    }
    source-packet-routing {
        adjacency-segment hold-time 240000;
        srgb start-label 800000 index-range 40000;
        explicit-null;
    }
    interface fxp0.0 {
        disable;
    }
    interface lo0.0 {
        passive;
    }
}
}

```

```

user@R4# show policy-options
policy-statement pplb {
    then {
        load-balance per-packet;
    }
}
policy-statement prefix-sid {

```

```

term 1 {
    from {
        route-filter 203.0.113.5/24 exact;
    }
    then {
        prefix-segment index 1004 node-segment;
        accept;
    }
}
}

```

```

user@R4# show routing-options
router-id 203.0.113.5;
forwarding-table {
    export pplb;
}

```

## Verification

### IN THIS SECTION

- [Verifying the IS-IS Adjacency Routes | 498](#)
- [Verifying the IS-IS Overview Information | 500](#)
- [Verifying the Segment Routing Route Entries for the IS-IS Protocol | 501](#)
- [Verifying the MPLS Segment Routing Route Entries for the IS-IS Protocol | 502](#)

Confirm that the configuration is working properly.

### Verifying the IS-IS Adjacency Routes

#### Purpose

Verify the adjacency of Router R4.



## Action

From operational mode, enter the `show isis adjacency detail` command.

```
user@R4> show isis adjacency detail

R5
  Interface: ge-0/0/0.0, Level: 1, State: Up, Expires in 25 secs
  Priority: 64, Up/Down transitions: 1, Last transition: 1d 23:55:22 ago
  Circuit type: 3, Speaks: IP, IPv6, MAC address: 0:5:86:e:2b:0
  Topologies: Unicast
  Restart capable: Yes, Adjacency advertisement: Advertise
  LAN id: R5.02, IP addresses: 10.0.6.2
  Level 1 IPv4 Adj-SID: 16

R5
  Interface: ge-0/0/0.0, Level: 2, State: Up, Expires in 25 secs
  Priority: 64, Up/Down transitions: 1, Last transition: 1d 23:55:22 ago
  Circuit type: 3, Speaks: IP, IPv6, MAC address: 0:5:86:e:2b:0
  Topologies: Unicast
  Restart capable: Yes, Adjacency advertisement: Advertise
  LAN id: R5.02, IP addresses: 10.0.6.2
  Level 2 IPv4 Adj-SID: 17

R0
  Interface: ge-0/0/1.0, Level: 1, State: Up, Expires in 7 secs
  Priority: 64, Up/Down transitions: 1, Last transition: 1d 23:49:06 ago
  Circuit type: 3, Speaks: IP, IPv6, MAC address: 0:5:86:5e:8e:1
  Topologies: Unicast
  Restart capable: Yes, Adjacency advertisement: Advertise
  LAN id: R1.02, IP addresses: 10.0.1.1
  Level 1 IPv4 Adj-SID: 18

R0
  Interface: ge-0/0/1.0, Level: 2, State: Up, Expires in 8 secs
  Priority: 64, Up/Down transitions: 1, Last transition: 1d 23:49:06 ago
  Circuit type: 3, Speaks: IP, IPv6, MAC address: 0:5:86:5e:8e:1
  Topologies: Unicast
  Restart capable: Yes, Adjacency advertisement: Advertise
  LAN id: R1.02, IP addresses: 10.0.1.1
  Level 2 IPv4 Adj-SID: 19
```

## Meaning

The output shows the IS-IS adjacency details of Router R4 with Router R0 and R5.

## Verifying the IS-IS Overview Information

### Purpose

Verify the IS-IS overview information of Router R4.

### Action

From operational mode, enter the `show isis overview` command.

```
user@R4> show isis overview
Instance: master
  Router ID: 203.0.113.5
  Hostname: R4
  Sysid: 0100.0404.0404
  Areaid: 47.0005.80ff.f800.0000.0108.0001
  Adjacency holddown: enabled
  Maximum Areas: 3
  LSP life time: 1200
  Attached bit evaluation: enabled
  SPF delay: 200 msec, SPF holddown: 5000 msec, SPF rapid runs: 3
  IPv4 is enabled, IPv6 is enabled, SPRING based MPLS is enabled
  Traffic engineering: enabled
  Restart: Disabled
    Helper mode: Enabled
  Layer2-map: Disabled
  Source Packet Routing (SPRING): Enabled
    SRGB Config Range:
      SRGB Start-Label : 800000, SRGB Index-Range : 40000
    SRGB Block Allocation: Success
      SRGB Start Index : 800000, SRGB Size : 40000, Label-Range: [ 800000, 839999 ]
  Node Segments: Disabled
Level 1
  Internal route preference: 15
  External route preference: 160
  Prefix export count: 0
  Wide metrics are enabled, Narrow metrics are enabled
  Source Packet Routing is enabled
```

```

Level 2
  Internal route preference: 18
  External route preference: 165
  Prefix export count: 0
  Wide metrics are enabled, Narrow metrics are enabled
  Source Packet Routing is enabled

```

## Meaning

The output displays the IS-IS overview information of the routing instance along with the SPRING details of Router R4.

## Verifying the Segment Routing Route Entries for the IS-IS Protocol

### Purpose

Verify the segment routing route entries of the routing table inet.3 for the IS-IS protocol.

### Action

From operational mode, enter the `show route table inet.3 protocol isis` command.

```

user@R4> show route table inet.3 protocol isis
inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

203.0.113.0/24    *[L-ISIS/14] 00:09:31, metric 10
                  to 10.0.6.2 via ge-0/0/0.0, Push 0
                  > to 10.0.1.1 via ge-0/0/1.0, Push 0
203.0.113.2/32   *[L-ISIS/14] 00:02:44, metric 20
                  > to 10.0.1.1 via ge-0/0/1.0, Push 801001

```

## Meaning

The output shows the segment routing routes of routing table inet.3 for the IS-IS protocol.

## Verifying the MPLS Segment Routing Route Entries for the IS-IS Protocol

### Purpose

Verify the MPLS segment routing route entries for the IS-IS protocol.

### Action

From operational mode, enter the `show route table mpls.0 protocol isis` command.

```
user@R4> show route table mpls.0 protocol isis

mpls.0: 23 destinations, 23 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0          *[MPLS/0] 2d 01:56:20, metric 1
            to table inet.0
0(S=0)     *[MPLS/0] 2d 01:56:20, metric 1
            to table mpls.0
1          *[MPLS/0] 2d 01:56:20, metric 1
            Receive
2          *[MPLS/0] 2d 01:56:20, metric 1
            to table inet6.0
2(S=0)     *[MPLS/0] 2d 01:56:20, metric 1
            to table mpls.0
13         *[MPLS/0] 2d 01:56:20, metric 1
            Receive
16         *[L-ISIS/14] 2d 01:52:56, metric 0
            > to 10.0.6.2 via ge-0/0/0.0, Pop
16(S=0)    *[L-ISIS/14] 00:01:34, metric 0
            > to 10.0.6.2 via ge-0/0/0.0, Pop
17         *[L-ISIS/14] 2d 01:52:56, metric 0
            > to 10.0.6.2 via ge-0/0/0.0, Pop
17(S=0)    *[L-ISIS/14] 00:10:49, metric 0
            > to 10.0.6.2 via ge-0/0/0.0, Pop
18         *[L-ISIS/14] 2d 01:46:40, metric 0
            > to 10.0.1.1 via ge-0/0/1.0, Pop
18(S=0)    *[L-ISIS/14] 00:01:34, metric 0
            > to 10.0.1.1 via ge-0/0/1.0, Pop
19         *[L-ISIS/14] 2d 01:46:40, metric 0
            > to 10.0.1.1 via ge-0/0/1.0, Pop
19(S=0)    *[L-ISIS/14] 00:10:49, metric 0
```

	> to 10.0.1.1 via ge-0/0/1.0, Pop
801000	*[L-ISIS/14] 2d 01:46:40, metric 10 to 10.0.6.2 via ge-0/0/0.0, Swap 801000 > to 10.0.1.1 via ge-0/0/1.0, Swap 0
801000(S=0)	*[L-ISIS/14] 00:01:34, metric 10 to 10.0.6.2 via ge-0/0/0.0, Swap 801000 > to 10.0.1.1 via ge-0/0/1.0, Pop
801001	*[L-ISIS/14] 2d 01:46:14, metric 10 to 10.0.6.2 via ge-0/0/0.0, Swap 801001 > to 10.0.1.1 via ge-0/0/1.0, Swap 801001
801002	*[L-ISIS/14] 1d 21:57:31, metric 10 to 10.0.6.2 via ge-0/0/0.0, Swap 801002 > to 10.0.1.1 via ge-0/0/1.0, Swap 801002
801003	*[L-ISIS/14] 1d 21:56:57, metric 10 to 10.0.6.2 via ge-0/0/0.0, Swap 801003 > to 10.0.1.1 via ge-0/0/1.0, Swap 801003
801005	*[L-ISIS/14] 2d 01:46:40, metric 10 to 10.0.6.2 via ge-0/0/0.0, Swap 0 > to 10.0.1.1 via ge-0/0/1.0, Swap 801005
801005(S=0)	*[L-ISIS/14] 00:01:34, metric 10 to 10.0.6.2 via ge-0/0/0.0, Pop > to 10.0.1.1 via ge-0/0/1.0, Swap 801005
801006	*[L-ISIS/14] 2d 01:46:40, metric 10 to 10.0.6.2 via ge-0/0/0.0, Swap 801006 > to 10.0.1.1 via ge-0/0/1.0, Swap 801006
801007	*[L-ISIS/14] 1d 21:56:24, metric 10 to 10.0.6.2 via ge-0/0/0.0, Swap 801007 > to 10.0.1.1 via ge-0/0/1.0, Swap 801007

Meaning

The output shows the MPLS segment routing route entries for protocol IS-IS.

RELATED DOCUMENTATION

<a href="#">Configure Anycast and Prefix segments in SPRING for IS-IS Protocol   506</a>
<a href="#">Configure Segment Routing Global Blocks Label Ranges in SPRING for IS-IS Protocol   504</a>
<a href="#">Example: Configuring SRGB in Segment Routing for IS-IS   475</a>
<a href="#">Overview of Adjacency Segments, Anycast Segments, and Configurable SRGB in SPRING   471</a>
<i>prefix-segment</i>

*source-packet-routing**srgb**traffic-engineering*

## Configure Segment Routing Global Blocks Label Ranges in SPRING for IS-IS Protocol

Segment routing (SR) or source packet routing in networking (SPRING) is a control-plane architecture that enables an ingress router to steer a packet through a specific set of nodes and links without relying on the intermediate nodes in the network to determine the actual path it should take. The label range for a segment routing global block (SRGB) is the range of label values used in segment routing. You can configure the start of the label range and the index range. The end of the label range is the summation of the start label value and the index range.

Before you configure SPRING SRGB for IS-IS protocol, you must:

- Configure the router interfaces.
- Configure ISIS.

To configure SPRING SRGB label range on a device:

1. Configure the start-label and index-range of SRGB. The start label value indicates the start of the SPRING label block and the index range along with the start label indicate the end of the label block.



### NOTE:

- Ensure that the MPLS label for a binding segment ID (SID) is the sum of the SRGB start label and SID index value. In addition, SID index value must be less than or equal to the index-range value specified in the configuration.
- Junos OS does not check whether the SID index is within the SRGB's range when the SID index is assigned through an ISIS export policy. If you configure an index that is out of range of the configured SRGB, you won't see any error message in the logs or while committing the configuration. Junos OS shows a commit error only when

you configure the SID under the **[edit protocols isis source-packet-routing]** hierarchy level.

```
[edit protocols isis source-packet-routing]
user@host# set srgb start-label start-label-value
user@host# set srgb index-range index-range-value
```



**NOTE:** The default value for the index range is 4096. This causes chunks of 256 label blocks being dynamically allocated by the label manager depending on the availability.

For example, configure SRGB with start-label 800,000 and index-range 40,000. The start label of the SPRING label block is 800,000 and the end of the label block is 840,000.

```
[edit protocols isis source-packet-routing]
user@host# set srgb start-label 800000
user@host# set srgb index-range 40000
```



**NOTE:** Ensure that the labels in the SRGB label range are not used by any other applications. If a label in the configured label range is used by another application, then a syslog error message RPD\_ISIS\_SRGBALLOCATIONFAIL is logged to indicate that the label manager is unable to allocate the requested SRGB label range. To free up the configured label range, check the label ranges configured at the **[edit protocol mpls label-range]** hierarchy level and re-configure the SRGB label range with a label range that is available and restart the routing protocol process (RPD).

## 2. Configure the value of IPv4 node segment index.

```
[edit protocols isis source-packet-routing]
user@host# set node-segment ipv4-index ipv4-index-value
```

For example, configure 1001 for IPv4 node segment index.

```
[edit protocols isis source-packet-routing]
user@host# set node-segment ipv4-index 1001
```

### 3. Configure the value of IPv6 node segment index.

```
[edit protocols isis source-packet-routing]
user@host# set node-segment ipv6-index ipv6-index-value
```

For example, configure 2001 for IPv6 node segment index.

```
[edit protocols isis source-packet-routing]
user@host# set node-segment ipv6-index 2001
```

## RELATED DOCUMENTATION

[Overview of Adjacency Segments, Anycast Segments, and Configurable SRGB in SPRING | 471](#)

[Example: Configuring SRGB in Segment Routing for IS-IS | 475](#)

*source-packet-routing*

## Configure Anycast and Prefix segments in SPRING for IS-IS Protocol

Segment routing (SR) or source packet routing in networking (SPRING) is a control-plane architecture that enables an ingress router to steer a packet through a specific set of nodes and links without relying on the intermediate nodes in the network to determine the actual path it should take. Segment routing global block (SRGB) is the range of label values used in segment routing. Junos OS allows you to configure prefix segment identifier (SID) and node SID to prefixes that are advertised in IS-IS through policy configuration.

Before you configure SPRING SRGB, prefix SID, and anycast SID for the IS-IS protocol, you must:

- Configure the router interfaces.
- Configure the router ID.
- Configure IS-IS.

To configure device R1 with SPRING SRGB, prefix SID, and anycast SID for IS-IS protocols:



1. Configure the start-label and index-range of SRGB.

```
[edit protocols isis source-packet-routing]
user@host# set srgb start-label start-label-value
user@host# set srgb index-range index-range-value
```

For example, configure SRGB with start-label 800000 and index-range 40000 .

```
[edit protocols isis source-packet-routing]
user@host# set srgb start-label 800000
user@host# set srgb index-range 40000
```

2. Configure the routing policy to match a route (IPv4 or IPv6 ) exactly. Configure the index and the node segment of the prefix segment for a given term and accept the routing policy.

```
[edit policy-options policy-statement policy-name term term-value]
user@host# set from route-filter IP address exact
user@host# set then prefix-segment index index-value
user@host# set then prefix-segment node-segment
user@host# set accept
```



**NOTE:** Configure node segment as /32 prefix on loopback interface (lo0.0) or on a valid stub interface.

For example, configure the routing policy to match the IPv4 route exactly. Configure the index and the node segment of the prefix segment for a given term and accept the routing policy.

```
[edit policy-options policy-statement policy-name term term-value]
user@host# set from route-filter 198.51.100.1/32 exact
user@host# set then prefix-segment index index-value
user@host# set then prefix-segment node-segment
user@host# set accept
```

For example, configure the routing policy to match the IPv6 route exactly. Configure the index and the node segment of the prefix segment for a given term and accept the routing policy.

```
[edit policy-options policy-statement policy-name term term-value]
user@host# set from route-filter 2001:db8::/32 exact
```

```

user@host# set then prefix-segment index index-value
user@host# set then prefix-segment node-segment
user@host# set accept

```

3. Configure the index and the node segment of the prefix segment for a given term and accept the routing policy.

```

[edit policy-options policy-statement policy-name term term-value then]
user@host# set prefix-segment index index-value
user@host# set prefix-segment node-segment
user@host# set accept

```

For example, configure the prefix segment with index 1004 and the node segment for term 1 of policy statement prefix SID and accept the routing policy.

```

[edit policy-options policy-statement prefix-sid term 1 then]
user@host# set prefix-segment index 1004
user@host# set prefix-segment node-segment
user@host# set accept

```

4. Configure the routing policy with the same prefix (IPv4 or IPv6 )and same prefix segment on more than one routers for anycast SID.



**NOTE:** For anycast prefix SID, configure prefix SID on loopback interface( lo0.0).

```

[edit policy-options policy-statement prefix-sid term 1 ]
user@host# set from route-filter IP address exact
user@host# set then prefix-segment index index-value
user@host# set then accept

```

For example, configure IPv4 prefix 198.51.100.1/32 with prefix segment 1000 on two routers R0 and R1 for anycast SID.

```

[edit policy-options policy-statement prefix-sid term 1 ]
user@host# set from route-filter 198.51.100.1/32 exact
user@host# set then prefix-segment index 1000
user@host# set then accept

```

For example, configure IPv6 prefix 2001:db8::/32 with prefix segment 1000 on two routers R0 and R1 for anycast SID.

```
[edit policy-options policy-statement prefix-sid term 1 ]
user@host# set from route-filter 2001:db8::/32 exact
user@host# set then prefix-segment index 2000
user@host# set then accept
```

5. Configure export policy on the IS-IS protocol.

```
[edit protocols isis]
user@host# export prefix-sid
```

6. Configure explicit NULL to enable E and P bits in all prefix SID advertisements.

```
[edit protocol isis source-packet-routing]
user@host# set explicit-null
```

7. Configure adjacency segment hold time to retain segment adjacency.

```
[edit protocol isis source-packet-routing]
user@host# set adjacency-segment hold-time hold-time
```

For example, configure adjacency segments with 240,000 milliseconds hold time.

```
[edit protocol isis source-packet-routing]
user@host# set adjacency-segment hold-time 240000
```

## RELATED DOCUMENTATION

[Overview of Adjacency Segments, Anycast Segments, and Configurable SRGB in SPRING | 471](#)

[source-packet-routing](#)

[traffic-engineering](#)

## Flexible Algorithms in IGP for Segment Routing

### SUMMARY

A flexible algorithm allows IGPs alone to compute constraint based paths over the network thereby providing simple traffic engineering without using a network controller. This is a light weight solution for networks that have not implemented a controller with full fledged segment routing but still want to reap the benefits of segment routing in their network.

### IN THIS SECTION

- [Understanding IGP Flexible Algorithms for Segment Routing | 510](#)

## Understanding IGP Flexible Algorithms for Segment Routing

### IN THIS SECTION

- [Benefits of Configuring Flexible Algorithm | 511](#)
- [What is Flexible Algorithm Definition \(FAD\)? | 511](#)
- [Participation in a Flexible Algorithm | 513](#)
- [Network Topology Configured with Flexible Algorithm Definitions | 514](#)
- [Flexible Algorithm RIBs | 518](#)
- [BGP Community and Flexible Algorithms | 518](#)
- [Application-specific Link Attribute based flexible algorithm | 519](#)
- [Strict Application-Specific Link Attribute based flexible algorithm | 519](#)
- [Supported and Unsupported Features | 520](#)

Define flexible algorithms that compute paths using different parameters and link constraints to thin slice a network based on your requirements. For example, you can define a flexible algorithm that computes a path to minimize IGP metric and define another flexible algorithm to compute a path based on traffic engineering metric to divide the network into separate planes. This feature allows networks without a controller to configure traffic engineering using segment routing without actually implementing a network controller. You can use the prefix SIDs to steer packets along the constraint-based paths. You can configure the prefix SIDs for flexible algorithm through policy configurations.

IGP protocols use a link metric to calculate a best path. However, the best IGP path might not always be the best path for certain types of traffic. Therefore, the IGP computed best path based on the shortest

IGP metric is often replaced with traffic engineered path due to the traffic requirements that are not reflected by the IGP metric. Typically RSVP-TE or SR TE is used for computing the path based on additional metrics and constraints to overcome this limitation. Junos installs such paths in the forwarding tables in addition to or as a replacement for the original path computed by the IGP.

### Benefits of Configuring Flexible Algorithm

- A lightweight version of segment routing traffic engineering that can be used in the core of the network.
- Allows you to configure traffic engineering using segment routing even without installing a network controller.
- Compute TI-LFA backup path using the same flexible algorithm definition and constraints computation.
- Ability to provision constrained primary path based on a single label.

### What is Flexible Algorithm Definition (FAD)?

A flexible algorithm allows IGP to calculate additional best paths based on specified constraints thereby providing simple traffic engineering without using a network controller. This is a lightweight solution for networks that have not implemented a controller with full fledged segment routing but still want to reap the benefits of segment routing in their network. Every operator can define separate constraints or colors depending on their requirements.

To define a flexible algorithm, include `flex-algorithm id` statement at the `[edit routing-options]` hierarchy level. The flexible algorithm definition (FAD) is assigned with an identifier ranging from 128 through 255. This flexible algorithm can be defined on one or more routers in a network. A flexible algorithm computes a best path based on the following parameters:

- **Calculation type**—SPF or strict SPF are the two available calculation type options. You can specify one of these calculation types in your FAD. Select the SPF calculation type if you want to influence the SPF computation on your device based on a certain local policy such as traffic engineering shortcuts. If you select strict SPF then the local policy cannot influence the SPF path selection.
- **Metric type**- IGP metric, TE metric, or delay metric are the available metric type options. You can specify one of these metric types in your FAD depending on your network requirement. If you do not want to use the IGP metric for a specific link you can configure a TE metric that IS-IS can use for calculating the route.
- **Priority**- You can assign a priority to your FADs as per your requirement and IS-IS prioritizes a particular FAD advertisement over another FAD based on your assigned priority.



**NOTE:** For FADs with link-constraints to work, all relevant links should advertise the admin-colors in IS-IS, which means either RSVP is enabled on the interfaces or set protocols isis traffic-engineering advertise always is configured.

- **Set of Link constraints-** You can configure admin-groups for many protocols at the [edit protocols mpls admin-groups] hierarchy level to color an individual link. These admin-groups can then be defined as include any, include-all or exclude at the [edit routing-options flex-algorithm *definition* admin-groups] hierarchy level.

We recommend configuring flexible algorithm definitions on only a few routers to provide redundancy and to avoid conflicts. Flexible algorithm definition is advertised in IGP as FAD sub-TLVs. In very large networks, we do not recommend configuring more than 8 flexible algorithms as each flexible algorithm will compute its own path and might cause performance issues beyond that.

It's also recommended that you configure multiple FAD servers in a specific ISIS Level before configuring any devices to participate in that FAD. In the case of an ISIS L1/L2 node (ABR), it's also recommended that you configure the FAD at both ISIS Level 1 and Level 2. If a FAD is configured only on a single ABR, traffic drops over flex algorithm paths are possible if the routing process restarts on that ABR. It's therefore a good design practice to have multiple ABRs, each of which has the FAD configured at both ISIS levels.

The default FAD has the following parameters:

- calculation type: spf
- metric type: igp-metric
- priority: 0
- Link constraints: none



**NOTE:** Modifying the flexible algorithm definition in a live network or on the fly could cause traffic disruptions until all the nodes converge on the new paths.

We support flexible Algorithm Definition (FAD) and Flexible Algorithm Prefix Metric (FAPM) in TED and implements two new corresponding TLVs "FAD TLV" and "FAPM TLV" in BGP-LS. The value of FAD TLV contains Flex-Algorithm, Metric-Type, Calculation-Type and Priority, all of which are one byte each. The TLV might have zero or more sub-TLVs included in it. The five sub-tlvs are Flex Algo Exclude Any Affinity, Flex Algo Include Any Affinity, Flex Algo Include All Affinity, Flex Algo Definition Flags and Flex Algo Exclude SRLG.

The FAD TLV can only be added to the BGP-LS Attribute of the Node NLRI if the corresponding node originates in the underlying IGP TLV or sub-TLV. The BGP-LS Attribute associated with a Node NLRI

might include one or more FAD TLVs corresponding to the Flexible Algorithm Definition for each algorithm that the node is advertising.

The value of FAPM TLV contains Flex-Algorithm (1 byte), Reserved (3 bytes) and Metric (4 bytes). The FAPM TLV can be added to the BGP-LS Attribute of the Prefix NLRI originated by a node, only if the corresponding node originates from the Prefix.

We've defined the Flexible Algorithm Prefix Metric (FAPM) to allow optimal end-to-end path for an interarea prefix. The area border router (ABR) must include the FAPM when advertising the prefix between areas that is reachable in that given Flexible Algorithm (flex algo). When a prefix is unreachable, the ABR must not include that prefix in that flex algo when advertising between areas. The defined FAPM provides inter-area support.

We support delay normalization and Flexible Algorithm Definition (FAD) defined constraints related to admin-groups and shared risk link group (SRLG) as defined in RFC 9350, IGP Flexible Algorithm.

During flexible algorithm computation, when the measured latency values are not equal and the difference is insignificant, IS-IS advertises this slightly higher latency value as a metric. IS-IS uses this normalized latency delay value instead of the measured delay value.

To configure flexible algorithm application specific SRLG values, include the application-specific statement at the `[edit protocols isis interface interface-name level level]` hierarchy level. To exclude SRLG constraint in an FAD, include the `exclude-srlg` statement at the `[edit routing-options flex-algorithm name definition]` hierarchy level.

You can control path selection by configuring the preference for OSPF Flexible Algorithm routes in `inetcolor.0` and `mpls.0` routing tables.

Configure `flex-algorithm-preference` statement at the `[edit protocols ospf]` hierarchy level to prioritize desired routes and improve traffic engineering across IP and MPLS domains.

## Participation in a Flexible Algorithm

You can configure specific routers to participate in a particular flexible algorithm as per your requirement. Paths computed based on a flexible algorithm definition is used by various applications each potentially using its own specific data plane for forwarding the data over such paths. The participating device must explicitly advertise its participation in a particular flexible algorithm to every application in the segment routing flexible algorithm sub TLV for IS-IS. You can configure a node to participate in a certain flexible algorithm provided it can support the constraints specified in that FAD.

To configure participation in a flexible algorithm include the `flex-algorithm` statement at the `[edit protocols isis source-packet- routing]` hierarchy level. The same device can advertise a FAD and also participate in a flexible algorithm.

Network Topology Configured with Flexible Algorithm Definitions

Figure 38 on page 514 shows the sample topology, there are 8 routers R0, R1, R2, R3, R4, R5, R6, and R7. Four flexible algorithms, 128, 129, 130, and 135 are defined and configured with admin-groups as listed in the following table:

Table 6: Flex Algorithm Definition (FAD) – Color Rules

Flex Algorithm Definition (FAD)	Color
128	Include any Red
129	Include any Green
130	Include any Green and Blue
135	Exclude Red

Figure 38: Flexible Algorithm Topology

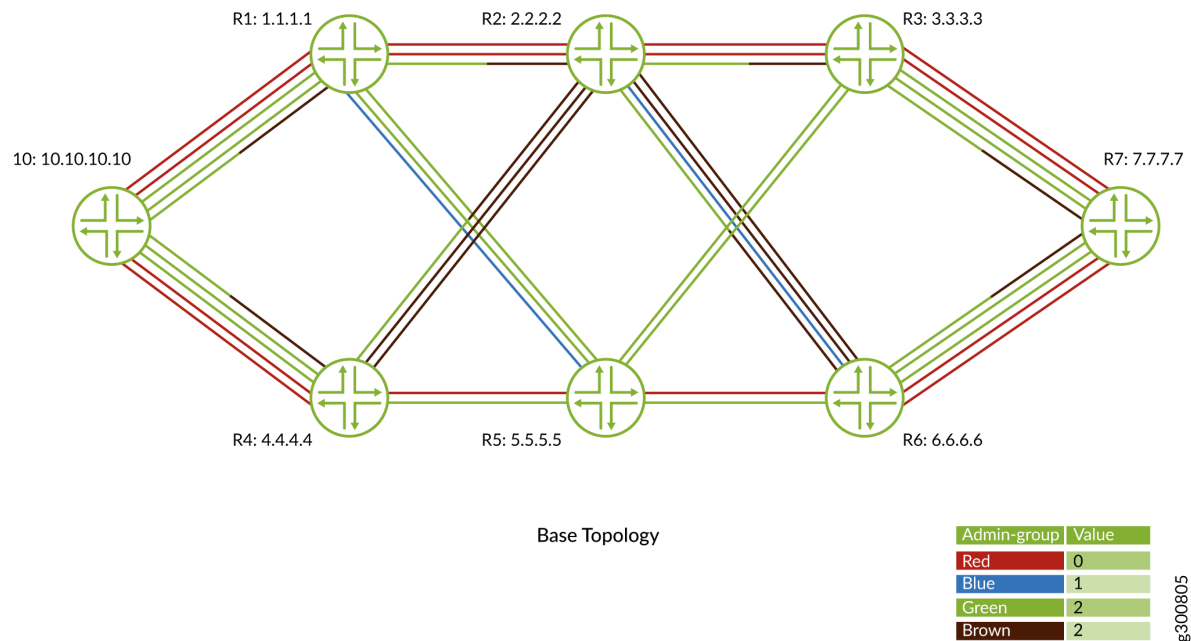




Figure 39 on page 515 shows how FAD 128 routes traffic on any interface that is configured with admin group red.

Figure 39: Traffic Flow for Flexible Algorithm Definition 128

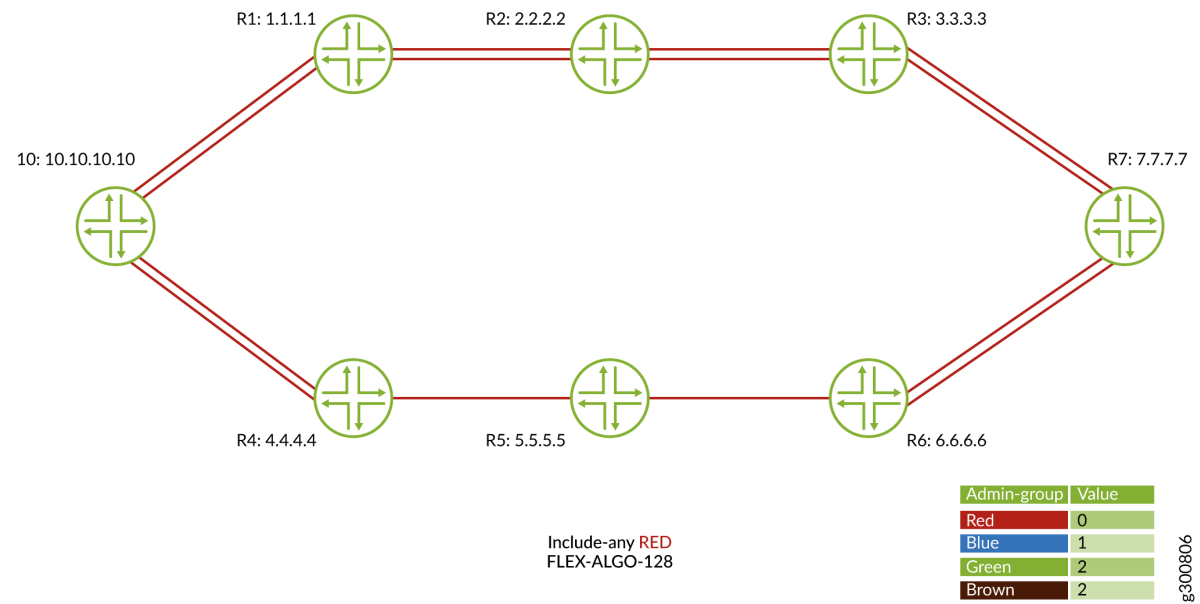


Figure 40 on page 516 shows how FAD 129 routes traffic on any interface that is configured with admin group green.

Figure 40: Traffic Flow for Flexible Algorithm Definition 129

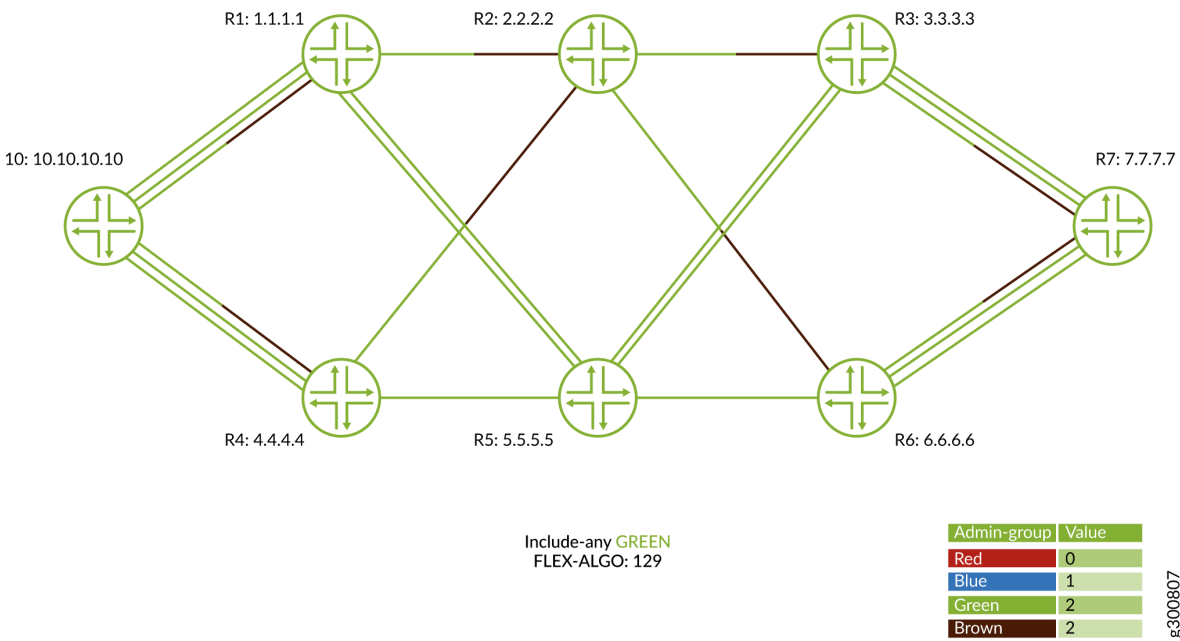


Figure 41 on page 517 shows how FAD 130 routes traffic on any interface that is configured with admin group green and blue.

Figure 41: Traffic flow for Flexible Algorithm Definition 130

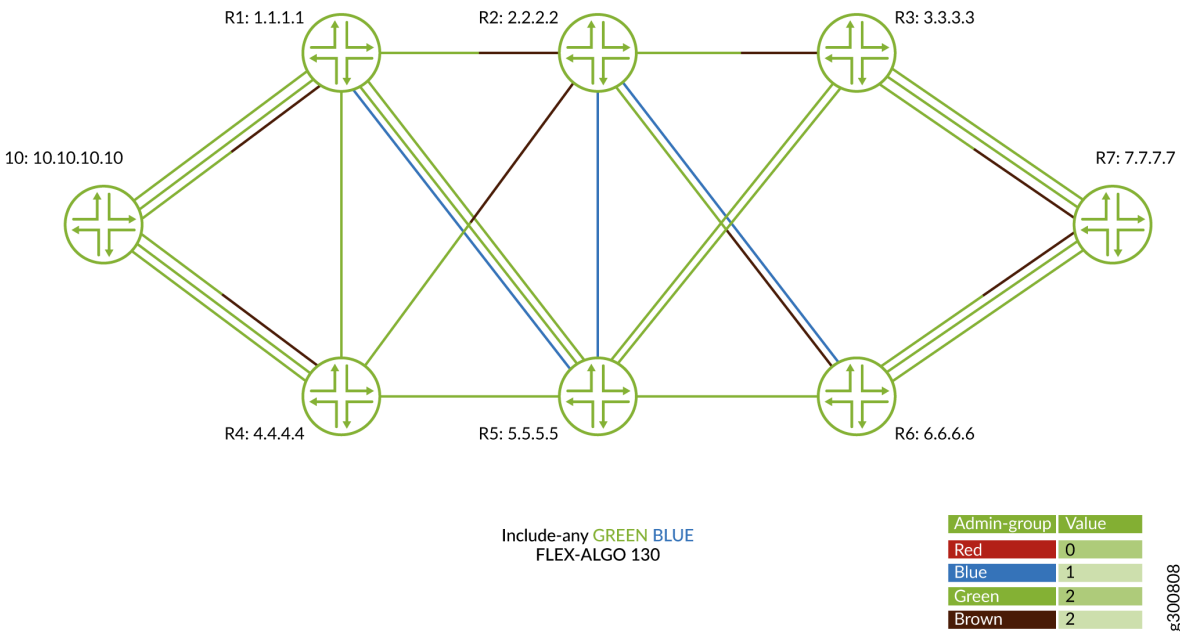
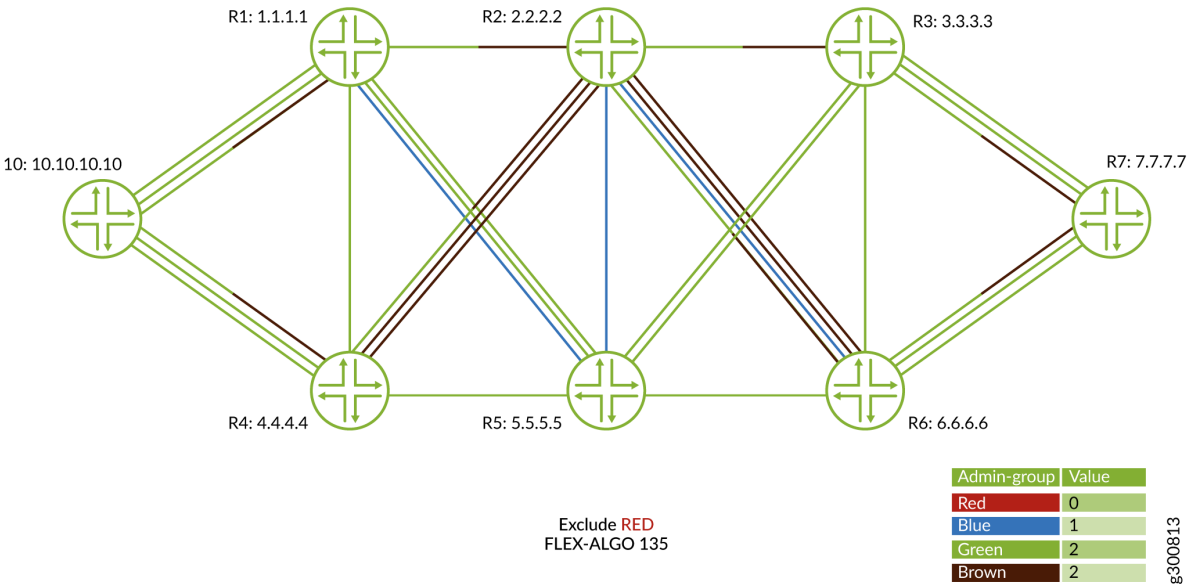


Figure 42 on page 518 shows how FAD 135 routes traffic on any interface that is not configured with admin group red.

Figure 42: Traffic Flow for Flexible Algorithm Definition 135



Flexible Algorithm RIBs

For every flexible algorithm that a router participates in, the corresponding flexible algorithm routes are installed in the corresponding flexible algorithm RIB groups also known as routing tables. By default, labeled flexible algorithm routes are installed in the `inet.color`, `inet(6)color.0` and `mpls.0` RIBs. They could also be installed in colored RIBs, such as `junos-rti-tc-<color>.inet(6).3` when `use-transport-class` statement is configured under `routing-options flex-algorithm <id>`. For more information, see [No Link Title No Link Title](#)

BGP Community and Flexible Algorithms

A flexible algorithm can be associated with a color. When a service prefix, such as a VPN service carries a BGP color extended community, by default the BGP service prefix resolves a flex-algo route that has the same associated color value. The flexible algorithm ingress routes that are installed in the `inet(6)color.0` tables will have this color value associated with the route. However, you can configure a different associate color value at the `[edit routing-options flex-algorithm id color color]` hierarchy level.

**NOTE:** Changing the associated color value in a flexible algorithm might result in traffic disruption. If you modify the color in a flexible algorithm definition, all routes pertaining to that flexible algorithm are removed from the RIB and added again with the new color.

You can leak BGP-LU prefixes into the IGP with flexible algorithm prefix-SIDs. For more information, see [Leaking BGP-LU Prefixes into Flexible Algorithm](#).

You can now leak BGP-CT prefixes into flexible algorithm and vice-versa. For more information, see [Leaking BGP-CT Prefixes into Flexible Algorithm](#).

### Application-specific Link Attribute based flexible algorithm

You can advertise different te-attributes such as te-metric, delay-metric, or admin-groups for RSVP and flexible algorithms on the same link. This is done using flexible algorithm specific application-specific link attribute as defined in RFC 8920.

The advantage of having a flexible algorithm application-specific link attribute advertise te-metric, delay-metric, or admin-groups is that a single link can advertise different te-link-attributes for legacy applications such as RSVP and different te-link-attributes for flexible algorithms.

To configure flexible algorithm application-specific te-attribute, include the application-specific statement at the [edit protocols ospf area interface] hierarchy level and the strict-asla-based-flex-algorithm statement at the [edit protocols ospf source-packet-routing] hierarchy level. With this implementation, it is no longer mandatory for the link to have RSVP enabled and [edit protocols ospf traffic-engineering advertisement always] to be configured which is the case with the existing behavior of advertising traffic engineering attributes.



**NOTE:** The Junos OS and Junos OS Evolved implementation of application-specific link attribute supports flexible algorithm applications only.

### Strict Application-Specific Link Attribute based flexible algorithm

The default behavior of application-specific flexible algorithm is to use the flexible algorithm application-specific te-attributes for a link if available, and if not, then fall back to the common application-specific te-attributes, and if neither are available, use the legacy te-attributes.

The configuration statement strict-asla-based-flex-algorithm at the [edit protocols ospf source-packet-routing] has to be applied to all the flexible algorithms running on the devices in the network to avoid routing loops.

If strict-asla-based-flex-algorithm is configured on all the devices, either a common application-specific te-attribute or flexible algorithm application-specific te-attribute must be advertised for each flexible algorithm link. In the absence of application-specific te-attributes, the device does not fall back to the legacy te-attributes and simply ignores the link.

The Operating System supports the following features in conjunction with application-specific link attribute based flexible algorithm:

- The application-specific te-attribute subTLV to comply with RFC 8920. The application-specific te-attributes sub-TLV is a sub-TLV of the OSPFv2 extended link TLV as defined in RFC 7684.
- Partially supports standard application identifier bit mask to advertise X-bit for flexible algorithms. Only the te-metric, delay-metric, or admin-groups are advertised as part of the application-specific link attribute sub-TLV.

The Operating System does not support the following features in conjunction with application-specific link attribute based flexible algorithm:

- Advertising user-defined application identifier bit masks is not supported.
- Readvertising flexible algorithm application-specific link attribute or rather any application-specific link attributes with BGP-LS is not supported because Traffic Engineering Database (TED) does not support application-specific link attribute.
- Advertising a common application-specific link attribute with standard application identifier bit mask and user-defined application identifier bit masks length set to zero is not supported.
- Advertising SRLG link constraint in flexible algorithm is not supported.
- Supporting traffic engineering for multiple applications is not supported, except for flexible algorithms.
- Defining admin-groups independent of MPLS is not supported.

### Supported and Unsupported Features

Junos OS supports flexible algorithms in the following scenarios:

- Support for configuring and advertising prefix SIDs for different flexible algorithms.
- Partially supports RFC 9350, *IGP Flexible Algorithm*
- Inter-level (IS-IS) leaking of flexible algorithm prefix SIDs is supported.
- The current implementation for flexible algorithms is supported for only OSPFv2 only as only OSPFv2 supports segment routing.

Junos OS does not support the following features in conjunction with flexible algorithms:

- Flexible algorithm is applicable only for default unicast topology, OSPFv2 multi-topology is not supported.
- IS-IS shortcuts and other IS-IS traffic engineering configuration options are not applicable for flexible algorithm computation
- Prefix and SID conflict resolution is not supported.

- Remote loop free alternate functionality is not supported because TI-LFA is the preferred FRR computation
- OSPFv2 shortcuts and other OSPFv2 traffic engineering configuration options are not applicable for flexible algorithm computation.
- Advertising flexible algorithm definition in the absence of flexible algorithm participation is not supported for OSPFv2.
- Extended Admin-Groups (EAG) are not supported because they are not supported in IS-IS.

SEE ALSO

<a href="#">Configuring Flexible Algorithm for Segment Routing   522</a>
<i>flex-algorithm</i>
<i>flex-algorithm</i>
<i>definition</i>
<i>show isis flex-algorithm</i>
<i>flex-algorithm (Protocols OSPF)</i>
<i>definition (Protocols OSPF)</i>
<i>application-specific (Protocols OSPF)</i>

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
25.4R1	Starting in Junos OS and Junos OS Evolved Release 25.4R1, you can control path selection by configuring the preference for OSPF Flexible Algorithm routes in <code>inetcolor.0</code> and <code>mpls.0</code> routing tables.
22.4R1	Starting in Junos OS Release 22.4R1, we've defined the Flexible Algorithm Prefix Metric (FAPM) to allow optimal end-to-end path for an interarea prefix.
22.2R1	Starting in Junos OS and Junos OS Evolved Release 22.2R1, you can advertise different te-attributes such as te-metric, delay-metric, or admin-groups for RSVP and flexible algorithms on the same link.

## Configuring Flexible Algorithm for Segment Routing

Before you begin configuring the flexible algorithm for IS-IS, make sure you:

1. Configure the device interfaces to enable IP transport.
2. Configure IS-IS protocol to enable dynamic routing protocol to exchange routing information.
3. Configure BGP protocol.
4. Configure segment routing.

To configure flexible algorithm for IS-IS:

1. Define flexible algorithm on routers that you have identified in your network. Assign an ID for the flexible algorithm definition (FAD) ranging from 128 through 255.

```
[edit routing-options]
user@host# set flex-algorithm id
```



**NOTE:** We recommend configuring flexible algorithm definition on only a few routers to avoid conflicts.

Specify the parameters of the definition. IS-IS calculates the path based on these specified parameters of the FAD.

- a. Map a BGP color community to the defined FAD. By default each flexible algorithm is associated with a value equal to the flex algorithm.

VPN can be made to resolve paths over the configured BGP color community.

```
[edit routing-options flex-algorithm id]
user@host# set color desired color community value
```



**NOTE:** Changing the BGP color community for a flexible algorithm might result in traffic disruption. If you modify a BGP color community for a flexible algorithm then all routes pertaining to that flexible algorithm are removed from the RIB and added again with new colors.



- b. Specify the calculation type based on which the IS-IS protocol calculates the path.

```
[edit routing-options flex-algorithm id definition]
user@host# set (spf | strict-spf)
```

- c. Specify the metric type based on which IS-IS calculates the path.

```
[edit routing-options flex-algorithm id definition]
user@host# set metric-type (delay-metric | igp-metric | te-metric)
```

- d. Assign a priority level to the advertisement of the FAD based on your requirement. Specify a priority ranging from 0 through 255.

```
[edit routing-options flex-algorithm id definition]
user@host# set priority priority
```



**NOTE:** Modifying the flexible algorithm definition could cause traffic disruptions until all the nodes converge on the new paths.

- e. If you have enabled RSVP traffic engineering, you can configure admin-groups for many protocols to color an individual link.

```
[edit protocols mpls]
user@host# set admin-groups
```

- f. Define the admin groups as per your requirement.

```
[edit routing-options flex-algorithm definition admin-group]
user@host# set include any admin-group
user@host# set include-all admin-group
user@host# set exclude admin-group
```



**NOTE:** For FADs with link-constraints to work, all relevant links should advertise the admin-colors in IS-IS. You must either enable RSVP on the interfaces or if you have

not configured RSVP for traffic engineering, make sure you configure **set traffic-engineering advertise always** at the [edit protocols isis] hierarchy level.

2. Identify the participating routers and configure participation on those routers. The same device can advertise a FAD and also participate in a flexible algorithm.

```
[edit protocols isis source-packet-routing]
user@host# set flex-algorithm id
```

3. Advertise prefix segments through policy configuration.

```
[edit policy-options policy-statement name term name]
user@host# set from route-filter route exact
user@host# set then prefix-segment algorithm id index value
user@host# set then prefix-segment algorithm id node-segment
```

4. Apply the policy under the protocol IS-IS.

```
[edit protocols isis]
user@host# set export name
```

5. To verify if your flexible algorithm configuration is working correctly use the `show isis spring flex-algorithm` command.

## Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS and OSPF

### IN THIS SECTION

- [Benefits of TI-LFA | 526](#)
- [Types of TI-LFA Protection | 526](#)
- [TI-LFA in IPv6 Networks | 527](#)
- [TI-LFA Limitations | 527](#)
- [Advertisement Flags for TI-LFA | 528](#)

Segment routing enables a router to send a packet along a specific path in the network by imposing a label stack that describes the path. The forwarding actions described by a segment routing label stack do not need to be established on a per-path basis. Therefore, an ingress router can instantiate an arbitrary path using a segment routing label stack and use it immediately without any signaling.

In segment routing, each node advertises mappings between incoming labels and forwarding actions. A specific forwarding action is referred to as a segment and the label that identifies that segment is referred to as a segment identifier (SID). The backup paths created by TI-LFA use the following types of segments:

- **Node segment**—A node segment forwards packets along the shortest path or paths to a destination node. The label representing the node segment (the node SID) is swapped until the destination node is reached.
- **Adjacency segment**—An adjacency segment forwards packets across a specific interface on the node that advertised the adjacency segment. The label representing an adjacency segment (the adjacency SID) is popped by the node that advertised it.

A router can send a packet along a specific path by creating a label stack that uses a combination of node SIDs and adjacency SIDs. Typically, node SIDs are used to represent parts of the path that correspond to the shortest path between two nodes. An adjacency SID is used wherever a node SID cannot be used to accurately represent the desired path.

Loop-free alternate (LFA) and remote LFA (RLFA) have been used to provide fast-reroute protection for several years. With LFA, a point of local repair (PLR) determines whether or not a packet sent to one of its direct neighbors reaches its destination without looping back through the PLR. In a typical network topology, approximately 40 to 60 percent of the destinations can be protected by LFA. Remote LFA expands on the concept of LFA by allowing the PLR to impose a single label to tunnel the packet to a repair tunnel endpoint from which the packet can reach its destination without looping back through the PLR. Using remote LFA, more destinations can be protected by the PLR compared to LFA. However, depending on the network topology, the percentage of destinations protected by remote LFA is usually less than 100 percent.

Topology-independent LFA (TI-LFA) extends the concept of LFA and remote LFA by allowing the PLR to use deeper label stacks to construct backup paths. In addition, the TI-LFA imposes the constraint that the backup path used by the PLR be the same path that a packet takes once the interior gateway protocol (IGP) has converged for a given failure scenario. This path is referred to as the post-convergence path.

Using the post-convergence path as the backup path has some desirable characteristics. For some topologies, a network operator only needs to make sure that the network has enough capacity to carry the traffic along the post-convergence path after a failure. In these cases, a network operator does not need to allocate additional capacity to deal with the traffic pattern immediately after the failure while the backup path is active, because the backup path follows the post-convergence path.

## Benefits of TI-LFA

- IGP automatically computes the backup path and the backup path follows the post failure path. You must plan capacity for post failure path and not allocate separate capacity for backup paths.
- Provides redundancy and protects against link failure.
- Easy to configure and utilize the post convergence path for transmission of packets.

## Types of TI-LFA Protection

TI-LFA provides protection against link failure, node failure, fate-sharing failures, and shared risk link group failures. In link failure mode, the destination is protected if the link fails. In node protection mode, the destination is protected if the neighbor connected to the primary link fails. To determine the node-protecting post-convergence path, the cost of all the links leaving the neighbor is assumed to increase by a configurable amount.

With fate-sharing protection, a list of fate-sharing groups are configured on each PLR with the links in each fate-sharing group identified by their respective IP addresses. The PLR associates a cost with each fate-sharing group. The fate-sharing-aware post-convergence path is computed by assuming that the cost of each link in the same fate-sharing group as the failed link has increased the cost associated with that group.

Configure Shared Risk Link Group (SRLG) protection in TI-LFA networks for segment routing to choose a fast reroute path that does not include SRLG links in the topology-independent loop-free alternate (TI-LFA) backup paths. SRLGs share a common fibre and they also share the risks of a broken link. When one link in an SRLG fails, other links in the group might also fail. Therefore, you need to avoid links that share the same risk as the protected link in the backup path. Configuring SRLG protection prevents TI-LFA from selecting backup paths that include a shared risk link. If you have configured SRLG protection then IS-IS computes the fast reroute path that is aligned with the post convergence path and excludes the links that belong to the SRLG of the protected link. All local and remote links that are from the same SRLG as the protected link are excluded from the TI-LFA back up path. The point of local repair (PLR) sets up the label stack for the fast reroute path with a different outgoing interface. Currently you cannot enable SRLG protection in IPv6 networks and in networks with multitopology.

In order to construct a backup path that follows the post-convergence path, TI-LFA uses several labels in the label stack that define the backup path. If the number of labels required to construct a particular post-convergence backup path exceeds a certain amount, it is useful in some circumstances to not install that backup path. You can configure the maximum number of labels that a backup path can have in order to be installed. The default value is 3, with a range of 2 through 5.

It is often the case that the post-convergence path for a given failure is actually a set of equal-cost paths. TI-LFA attempts to construct the backup paths to a given destination using multiple equal-cost paths in the post-failure topology. Depending on the topology, TI-LFA might need to use different label stacks to accurately construct those equal-cost backup paths. By default, TI-LFA only installs one

backup path for a given destination. However, you can configure the value in the range from 1 through 8.

## TI-LFA in IPv6 Networks

Configure TI-LFA with segment routing in an IPv6-only network to provide fast reroute (FRR) backup paths corresponding to the post-convergence path for a given failure. However, you cannot configure fate-sharing protection for IPv6-only networks. To compute backup paths in IPv6-only networks, the IS-IS protocol must advertise the following TLV types:

- TLV 233 - IPv6 Global Interface Address
- Subtlv 12 and 13 of TLV 22

You can configure multiple global IPv6 addresses on the interface. If you configure the `use-source-packet-routing` statement, then all addresses are assigned TI-LFA backup paths.

Configure a point of local repair (PLR) to create a topology independent loop-free alternate backup path for prefix-SIDs derived from Segment Routing Mapping Server advertisements in an IS-IS network. In a network configured with segment routing, IS-IS uses the Segment Routing Mapping Server advertisements to derive prefix-SIDs. Segment Routing Mapping Server advertisements for IPv6 are currently not supported. To attach flags to Segment Routing Mapping Server advertisements, include the `attached`, `domain-wide-flooding`, and `no-node-segment` statements at the `[edit routing-options source-packet-routing mapping-server-entry mapping-server-name]` hierarchy level.

## TI-LFA Limitations

The backup path for prefix-SIDs from Segment Routing Mapping Server advertisements are not created in the following scenarios:

- If some hops are present in a non-SR domain.
- If the segment routing node is advertising a prefix and a prefix-SID index directly, then Junos OS uses the prefix-SID index and disregards the mapping server advertisement for that prefix.
- If a backup path requires an adjacency-SID from the LDP domain then the backup path cannot be installed.
- If the PLR is unable to determine the label mapping using LDP.



**NOTE:** Currently you cannot configure remote LFA and TI-LFA on a SR-LDP stitching node in the same instance. Therefore, you cannot configure both `post-convergence-lfa` and `link-protection` on the same device.

## Advertisement Flags for TI-LFA

Set the following mapping server advertisement flags to indicate the origin of the advertised prefix:

Flag	TLV Name	Flag Values	Length	Description
A	Label Binding TLV	0, 1 default value is 0	1	Attached Flag—Include the attached configuration statement to set this flag to 1 to indicate that the prefixes and SIDs advertised in the SID or Label Binding TLV are directly connected to their originators.
S	Label Binding TLV	0, 1 default value is 0	1	Include the domain-wide-flooding configuration statement to set this flag to 1 to indicate that the SID or Label Binding TLV is flooded across the entire routing domain.
D	Label Binding TLV	0, 1 default value is 0	1	Set by a border node when readvertising a SID or Label Binding TLV to indicate that the SID or Label Binding TLV is leaked from level 2 to level 1.
N	Prefix-SID sub TLV	0, 1 default value is 1	1	Include the no-node-segment configuration statement to set this flag to 0 to indicate that the prefix has originated from a single node.

## RELATED DOCUMENTATION

[Configuring Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS | 529](#)

[Example: Configuring Topology Independent Loop-Free Alternate with Segment Routing for IS-IS | 531](#)

*post-convergence-lfa*

*use-post-convergence-lfa*

*use-for-post-convergence-lfa*

*node-protection*

## Configuring Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS

Loop-free alternate (LFA) and remote LFA have been used to provide fast-reroute protection for several years. With LFA, a point of local repair (PLR) determines whether or not a packet sent to one of its direct neighbors will reach its destination without looping back through the PLR. In a typical network topology, perhaps 40-60 percent of destinations can be protected by LFA. Remote LFA expands on the concept of LFA by allowing the PLR to impose a single label to tunnel the packet to a repair tunnel endpoint from which the packet can reach its destination without looping back through the PLR. Using remote LFA, more destinations can be protected by the PLR compared to LFA. However, depending on the network topology, the percentage of destinations protected by remote LFA usually less than 100 percent.

Topology-independent loop-free alternate (TI-LFA) extends the concept of LFA and remote LFA by allowing the PLR to use deeper label stacks to construct backup paths. In addition, TI-LFA imposes the constraint that the backup path used by the PLR be the same path that a packet takes once the IGP converges for a given failure scenario. This path is referred to as the post-convergence path.

Using the post-convergence path as the backup path has some desirable characteristics. For some topologies, a network operator only needs to make sure that the network has enough capacity to carry the traffic along the post-convergence path after a failure. In these cases, a network operator does not need to allocate additional capacity to deal with the traffic pattern immediately after the failure while the backup path is active, because the backup path follows the post-convergence path.

Before you configure TI-LFA for IS-IS, be sure you configure SPRING or segment routing.

To configure TI-LFA using SPRING for IS-IS, you must do the following:

1. Enable TI-LFA for IS-IS protocol.

```
[edit protocols isis backup-spf-options]
user@R1# set use-post-convergence-lfa
```

2. (Optional) Configure backup shortest path first (SPF) attributes such as maximum equal-cost multipath (ECMP) backup paths and maximum labels for TI-LFA for the IS-IS protocol.

```
[edit protocols isis backup-spf-options use-post-convergence-lfa]
user@R1# set maximum-backup-paths maximum-backup-paths
user@R1# set maximum-labels maximum-labels
```

3. Configure the computation and installation of a backup path that follows the post-convergence path on the given interface and level for the IS-IS protocol.

```
[edit protocols isis interface interface-name level level
user@R1# set post-convergence-lfa
```

4. (Optional) Enable fate-sharing protection for a given interface and level. Specify the fate-sharing group to use as a constraint for the post-convergence path.



**NOTE:** You do not have to configure the `use-for-post-convergence-lfa` statement and the `fate-sharing-protection` statement for basic link protection for the backup path.

```
[edit routing-options fate-sharing group group-name]
user@R1# set use-for-post-convergence-lfa
```

```
[edit protocols isis interface interface-name level level post-convergence-lfa]
user@R1# set fate-sharing-protection
```

5. (Optional) Enable node protection for a given interface and level.

```
[edit protocols isis interface interface-name level level post-convergence-lfa]
user@R1# set node-protection
```

## RELATED DOCUMENTATION

[Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS and OSPF | 524](#)

[Example: Configuring Topology Independent Loop-Free Alternate with Segment Routing for IS-IS | 531](#)

*post-convergence-lfa*

*use-post-convergence-lfa*

*use-for-post-convergence-lfa*

*node-protection*



## Example: Configuring Topology Independent Loop-Free Alternate with Segment Routing for IS-IS

### IN THIS SECTION

- Requirements | 531
- Overview | 532
- Configuration | 533

This example shows topology-independent loop-free alternate (TI-LFA) with segment routing for the IS-IS protocol to provide MPLS fast reroute (FRR) backup paths corresponding to the post-convergence path for a given failure by using deeper label stacks to construct backup paths. TI-LFA provides protection against link failure, node failure, and fate-sharing failures. In link failure mode, the destination is protected if the link fails. In node protection mode, the destination is protected if the neighbor connected to the primary link fails. To determine the node-protecting post-convergence path, the cost of all the links leaving the neighbor is assumed to increase by a configurable amount. With fate-sharing protection, a list of fate-sharing groups are configured on each PLR with the links in each fate-sharing group identified by their respective IP addresses.



**NOTE:** Our content testing team has validated and updated this example.

### Requirements

This example uses the following hardware and software components:

- Nine MX Series routers
- Junos OS Release 17.4 or later running on all devices
  - Updated and revalidated using vMX on Junos OS Release 21.1R1.

Before you configure TI-LFA routes using SPRING for IS-IS, be sure you configure SPRING or segment routing.



**NOTE:** Are you interested in getting hands-on experience on this feature?

Visit Juniper vLabs to reserve your pre-configured [vLab Sandbox: Segment Routing - Basic](#) and try it out for free!

## Overview

### IN THIS SECTION

- [Topology | 533](#)

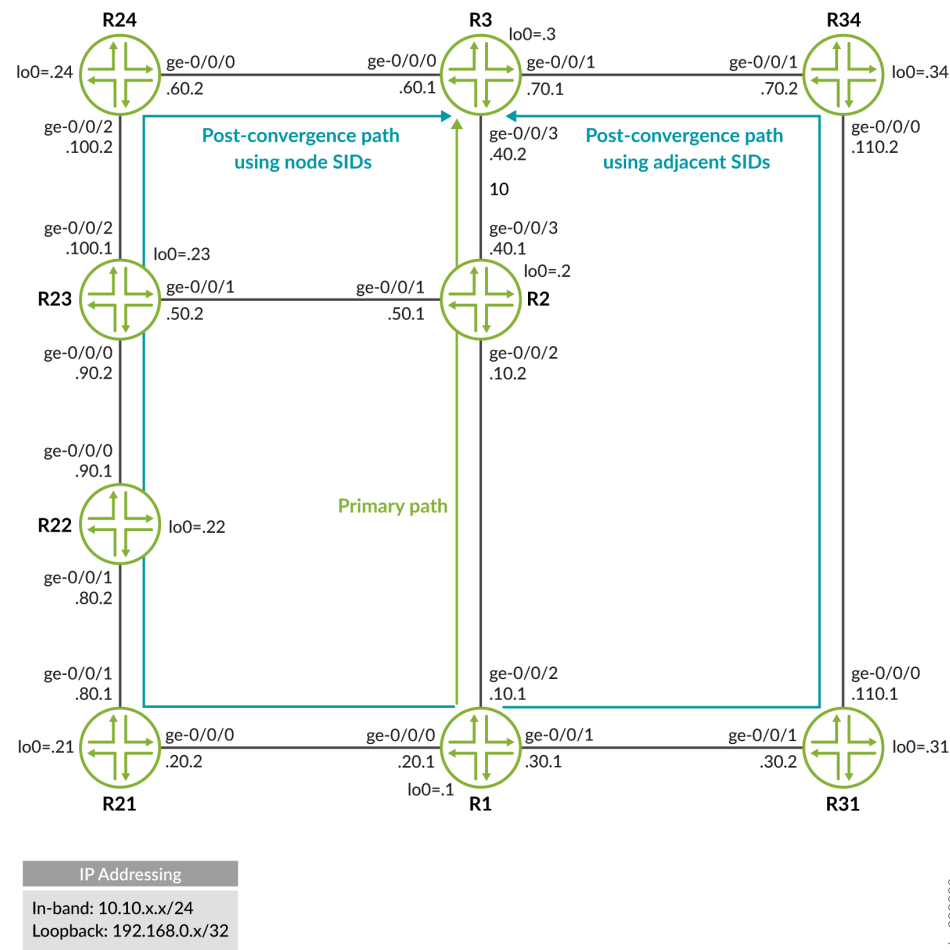
Junos OS allows you to enable TI-LFA for IS-IS by configuring the `use-post-convergence-lfa` statement at the `[edit protocols isis backup-spf-options]` hierarchy level. You can enable the creation of post-convergence backup paths for a given interface by configuring the `post-convergence-lfa` statement at the `[edit protocols isis interface interface-name level level]` hierarchy level.

TI-LFA provides protection against link failure, node failure, and failures of fate-sharing groups. You can enable link-protection mode using the `post-convergence-lfa` statement. You can enable node-protection mode, or fate-sharing-protection mode, or both modes, for a given interface at the `[edit protocols isis interface interface-name level level post-convergence-lfa]` hierarchy level. To ensure that the fate-sharing protection is enabled for a given fate-sharing group, you need to configure the `use-for-post-convergence-lfa` statement at the `[edit routing-options fate-sharing group group-name]` hierarchy level.



**NOTE:** TI-LFA supports protection of routes for both IPv4 and IPv6 prefixes. This example demonstrates protection of routes for IPv4 prefixes.

Topology



Configuration

IN THIS SECTION

- [Verification](#) | 546

CLI Quick Configuration

To quickly configure link-protection in this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration,

copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter commit from configuration mode.

## R1

```

set interfaces ge-0/0/0 unit 0 description r1-to-r21
set interfaces ge-0/0/0 unit 0 family inet address 10.10.20.1/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/1 unit 0 description r1-to-r31
set interfaces ge-0/0/1 unit 0 family inet address 10.10.30.1/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces ge-0/0/2 unit 0 description r1-to-r2
set interfaces ge-0/0/2 unit 0 family inet address 10.10.10.1/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/2 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 198.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0000.2222.0001.00
set interfaces lo0 unit 0 family mpls
set protocols isis interface ge-0/0/0.0 level 2 disable
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 disable
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/2.0 level 2 disable
set protocols isis interface ge-0/0/2.0 level 1 post-convergence-lfa
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing node-segment ipv4-index 1001
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols mpls interface ge-0/0/0.0
set protocols mpls interface ge-0/0/1.0
set protocols mpls interface ge-0/0/2.0
set policy-options policy-statement pplb then load-balance per-packet
set routing-options forwarding-table export pplb
set routing-options router-id 198.168.0.1

```

**R2**

```

set interfaces ge-0/0/1 unit 0 description r2-to-r23
set interfaces ge-0/0/1 unit 0 family inet address 10.10.50.1/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces ge-0/0/2 unit 0 description r2-to-r1
set interfaces ge-0/0/2 unit 0 family inet address 10.10.10.2/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/2 unit 0 family mpls
set interfaces ge-0/0/3 unit 0 description r2-to-r3
set interfaces ge-0/0/3 unit 0 family inet address 10.10.40.1/24
set interfaces ge-0/0/3 unit 0 family iso
set interfaces ge-0/0/3 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0000.2222.0002.00
set interfaces lo0 unit 0 family mpls
set protocols isis interface ge-0/0/1.0 level 2 disable
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/2.0 level 2 disable
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface ge-0/0/3.0 level 2 disable
set protocols isis interface ge-0/0/3.0 point-to-point
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing node-segment ipv4-index 1002
set protocols mpls interface ge-0/0/1.0
set protocols mpls interface ge-0/0/2.0
set protocols mpls interface ge-0/0/3.0
set routing-options router-id 192.168.0.2

```

**R3**

```

set interfaces ge-0/0/0 unit 0 description r3-to-r24
set interfaces ge-0/0/0 unit 0 family inet address 10.10.60.1/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/1 unit 0 description r3-to-r34
set interfaces ge-0/0/1 unit 0 family inet address 10.10.70.1/24

```

```

set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces ge-0/0/3 unit 0 description r3-to-r2
set interfaces ge-0/0/3 unit 0 family inet address 10.10.40.2/24
set interfaces ge-0/0/3 unit 0 family iso
set interfaces ge-0/0/3 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0000.2222.0003.00
set interfaces lo0 unit 0 family mpls
set protocols isis interface ge-0/0/0.0 level 2 disable
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 disable
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/3.0 level 2 disable
set protocols isis interface ge-0/0/3.0 point-to-point
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing node-segment ipv4-index 1003
set protocols mpls interface ge-0/0/0.0
set protocols mpls interface ge-0/0/1.0
set protocols mpls interface ge-0/0/3.0
set routing-options router-id 192.168.0.3

```

## R21

```

set interfaces ge-0/0/0 unit 0 description r21-to-r1
set interfaces ge-0/0/0 unit 0 family inet address 10.10.20.2/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/1 unit 0 description r21-to-r22
set interfaces ge-0/0/1 unit 0 family inet address 10.10.80.1/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.21/32
set interfaces lo0 unit 0 family iso address 49.0000.2222.0021.00
set interfaces lo0 unit 0 family mpls
set protocols isis interface ge-0/0/0.0 level 2 disable
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 disable
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface fxp0.0 disable

```

```

set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing node-segment ipv4-index 1021
set protocols mpls interface ge-0/0/0.0
set protocols mpls interface ge-0/0/1.0
set routing-options router-id 192.168.0.21

```

## R22

```

set interfaces ge-0/0/0 unit 0 description r22-to-r23
set interfaces ge-0/0/0 unit 0 family inet address 10.10.90.1/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/1 unit 0 description r22-to-r21
set interfaces ge-0/0/1 unit 0 family inet address 10.10.80.2/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.22/32
set interfaces lo0 unit 0 family iso address 49.0000.2222.0022.00
set interfaces lo0 unit 0 family mpls
set protocols isis interface ge-0/0/0.0 level 2 disable
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 disable
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing node-segment ipv4-index 1022
set protocols mpls interface ge-0/0/0.0
set protocols mpls interface ge-0/0/1.0
set routing-options router-id 192.168.0.22

```

## R23

```

set interfaces ge-0/0/0 unit 0 description r23-to-r22
set interfaces ge-0/0/0 unit 0 family inet address 10.10.90.2/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/1 unit 0 description r23-to-r2
set interfaces ge-0/0/1 unit 0 family inet address 10.10.50.2/24
set interfaces ge-0/0/1 unit 0 family iso

```

```

set interfaces ge-0/0/1 unit 0 family mpls
set interfaces ge-0/0/2 unit 0 description r23-to-r24
set interfaces ge-0/0/2 unit 0 family inet address 10.10.100.1/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/2 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.23/32
set interfaces lo0 unit 0 family iso address 49.0000.2222.0023.00
set interfaces lo0 unit 0 family mpls
set protocols isis interface ge-0/0/0.0 level 2 disable
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 disable
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/2.0 level 2 disable
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing node-segment ipv4-index 1023
set protocols mpls interface ge-0/0/0.0
set protocols mpls interface ge-0/0/1.0
set protocols mpls interface ge-0/0/2.0
set routing-options router-id 192.168.0.23

```

## R24

```

set interfaces ge-0/0/0 unit 0 description r24-to-r3
set interfaces ge-0/0/0 unit 0 family inet address 10.10.60.2/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/2 unit 0 description r24-to-r23
set interfaces ge-0/0/2 unit 0 family inet address 10.10.100.2/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/2 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.24/32
set interfaces lo0 unit 0 family iso address 49.0000.2222.0024.00
set interfaces lo0 unit 0 family mpls
set protocols isis interface ge-0/0/0.0 level 2 disable
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/2.0 level 2 disable
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive

```



```

set protocols isis source-packet-routing node-segment ipv4-index 1024
set protocols mpls interface ge-0/0/0.0
set protocols mpls interface ge-0/0/2.0
set routing-options router-id 192.168.0.24

```

### R31

```

set interfaces ge-0/0/0 unit 0 description r31-to-r34
set interfaces ge-0/0/0 unit 0 family inet address 10.10.110.1/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/1 unit 0 description r31-to-r1
set interfaces ge-0/0/1 unit 0 family inet address 10.10.30.2/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 198.162.0.31/32
set interfaces lo0 unit 0 family iso address 49.0000.2222.0031.00
set interfaces lo0 unit 0 family mpls
set protocols isis interface ge-0/0/0.0 level 1 metric 500
set protocols isis interface ge-0/0/0.0 level 2 disable
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 1 metric 10
set protocols isis interface ge-0/0/1.0 level 2 disable
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing node-segment ipv4-index 1031
set protocols mpls interface ge-0/0/1.0
set protocols mpls interface ge-0/0/0.0
set routing-options router-id 198.162.0.31

```

### R34

```

set interfaces ge-0/0/0 unit 0 description r34-to-r31
set interfaces ge-0/0/0 unit 0 family inet address 10.10.110.2/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/1 unit 0 description r34-to-r3
set interfaces ge-0/0/1 unit 0 family inet address 10.10.70.2/24

```

```

set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 192.168.0.34/32
set interfaces lo0 unit 0 family iso address 49.0000.2222.0034.00
set interfaces lo0 unit 0 family mpls
set protocols isis interface ge-0/0/0.0 level 1 metric 500
set protocols isis interface ge-0/0/0.0 level 2 disable
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 1 metric 10
set protocols isis interface ge-0/0/1.0 level 2 disable
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing node-segment ipv4-index 1034
set protocols mpls interface ge-0/0/0.0
set protocols mpls interface ge-0/0/1.0
set routing-options router-id 192.168.0.34

```

## Configuring R1

### Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [Junos OS CLI User Guide](#).

To configure Device R1:

1. Configure the interfaces.

```

[edit interfaces]
user@R1# set ge-0/0/0 unit 0 description r1-to-r21
user@R1# set ge-0/0/0 unit 0 family inet address 10.10.20.1/24
user@R1# set ge-0/0/0 unit 0 family iso
user@R1# set ge-0/0/0 unit 0 family mpls

user@R1# set ge-0/0/1 unit 0 description r1-to-r31
user@R1# set ge-0/0/1 unit 0 family inet address 10.10.30.1/24
user@R1# set ge-0/0/1 unit 0 family iso
user@R1# set ge-0/0/1 unit 0 family mpls

user@R1# set ge-0/0/2 unit 0 description r1-to-r2

```

```

user@R1# set ge-0/0/2 unit 0 family inet address 10.10.10.1/24
user@R1# set ge-0/0/2 unit 0 family iso
user@R1# set ge-0/0/2 unit 0 family mpls

user@R1# set lo0 unit 0 family inet address 198.168.0.1/32
user@R1# set lo0 unit 0 family iso address 49.0000.2222.0001.00
user@R1# set lo0 unit 0 family mpls

```

## 2. Configure the router ID.

```

[edit routing-options]
user@R1# set router-id 198.168.0.1

```

## 3. Configure MPLS.

```

[edit protocols]
user@R1# set mpls interface ge-0/0/0.0
user@R1# set mpls interface ge-0/0/1.0
user@R1# set mpls interface ge-0/0/2.0

```

## 4. Configure IS-IS.

```

[edit protocols]
user@R1# set isis interface ge-0/0/0.0 level 2 disable
user@R1# set isis interface ge-0/0/0.0 point-to-point

user@R1# set isis interface ge-0/0/1.0 level 2 disable
user@R1# set isis interface ge-0/0/1.0 point-to-point

user@R1# set isis interface ge-0/0/2.0 level 2 disable
user@R1# set isis interface ge-0/0/2.0 point-to-point

user@R1# set isis interface lo0.0 passive

user@R1# set isis interface fxp0.0 disable

```

5. Configure to install backup route along the link-protecting post-convergence path on interface ge-0/0/2.

```
[edit protocols]
user@R1# set isis interface ge-0/0/2.0 level 1 post-convergence-lfa
```

6. Configure the maximum number of labels for segment routing routed paths for protection of backup shortest-path-first attributes.

```
[edit protocols]
user@R1# set isis backup-spf-options use-post-convergence-lfa maximum-labels 8
```

7. Configure IPv4 index and index range for node segments in segment routing for the IS-IS protocol.

```
[edit protocols]
user@R1# set isis source-packet-routing node-segment ipv4-index 1001
```

8. (Optional) Enable node-protection on interface ge-0/0/2.

```
[edit protocols]
user@R1# set isis interface ge-0/0/2 level 2 post-convergence-lfa node-protection cost 2000
```

9. (Optional) Configure the fate-sharing group cost.

```
[edit routing-options]
user@R1# set fate-sharing group fs-group-1 cost 3000
```

10. (Optional) Configure the fate-sharing group to indicate that link from Device R1 to Device R2 and the link from Device R21 to Device R22 share fate and allow it to be used for post-convergence-lfa.

```
[edit routing-options]
user@R1# set fate-sharing group fs-group-1 from 10.10.10.1 to 10.10.10.2
user@R1# set fate-sharing group fs-group-1 from 10.10.80.1 to 10.10.80.2
user@R1# set fate-sharing group fs-group-1 use-for-post-convergence-lfa
```

11. (Optional) Enable fate-sharing protection for ge-0/0/2 on Device R1.

```
[edit protocols]
user@R1# set isis interface ge-0/0/2 level 2 post-convergence-lfa fate-sharing-protection
```

12. Configure a per packet load-balance policy for TI-LFA to work and ensure faster convergence.

```
[edit]
user@R1# set policy-options policy-statement pplb then load-balance per-packet
```

13. Apply the policy to export the routes into the forwarding table.

```
[edit]
user@R1# set routing-options forwarding-table export pplb
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols` and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces
ge-0/0/0 {
  unit 0 {
    description r1-to-r21;
    family inet {
      address 10.10.20.1/24;
    }
    family iso;
    family mpls;
  }
}
ge-0/0/1 {
  unit 0 {
    description r1-to-r31;
    family inet {
      address 10.10.30.1/24;
    }
    family iso;
```

```

        family mpls;
    }
}
ge-0/0/2 {
    unit 0 {
        description r1-to-r2;
        family inet {
            address 10.10.10.1/24;
        }
        family iso;
        family mpls;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 198.168.0.1/32;
        }
        family iso {
            address 49.0000.2222.0001.00;
        }
        family mpls;
    }
}

```

```

user@R1# show routing-options
router-id 198.168.0.1;
forwarding-table {
    export pplb;
}

```

```

user@R1# show policy-options
policy-statement pplb {
    then {
        load-balance per-packet;
    }
}

```

```

    }
}

```

```

user@R1# show protocols
isis {
    interface ge-0/0/0.0 {
        level 2 disable;
        point-to-point;
    }
    interface ge-0/0/1.0 {
        level 2 disable;
        point-to-point;
    }
    interface ge-0/0/2.0 {
        level 2 disable;
        level 1 {
            post-convergence-lfa;
        }
        point-to-point;
    }
    interface fxp0.0 {
        disable;
    }
    interface lo0.0 {
        passive;
    }
    source-packet-routing {
        node-segment ipv4-index 1001;
    }
    backup-spf-options {
        use-post-convergence-lfa maximum-labels 8;
    }
}
mpls {
    interface ge-0/0/0.0;
    interface ge-0/0/1.0;
    interface ge-0/0/2.0;
}

```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Verify the TI-LFA routes using node SIDs | 546](#)
- [Verify adjacency SIDs | 547](#)
- [Verify the TI-LFA routes using adjacency SIDs | 548](#)

Confirm that the configuration is working properly.

### *Verify the TI-LFA routes using node SIDs*

#### Purpose

Verify the link-protecting backup path for primary next hops on interface ge-0/0/2 for Device R1 and verify if the backup path to reach 192.168.0.3/32 has been created and has the correct label stack.

#### Action

From operational mode, run the `show route 192.168.0.3` command to display the routing table information.

```
user@R1> show route 192.168.0.3

inet.0: 38 destinations, 38 routes (38 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.0.3/32    *[IS-IS/15] 09:52:56, metric 20
                  > to 10.10.10.2 via ge-0/0/2.0

inet.3: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.0.3/32    *[L-ISIS/14] 05:45:40, metric 20
                  > to 10.10.10.2 via ge-0/0/2.0, Push 801003
                  to 10.10.20.2 via ge-0/0/0.0, Push 801003, Push 801024(top)
```



## Meaning

The primary path to reach 198.168.0.3/32 (corresponding to Device R3) is through the interface ge-0/0/2 with a label of 801003, corresponding to the node-SID of Device R3. If the interface ge-0/0/2 fails, the backup path using the interface ge-0/0/0 using the label stack [801024, 801003] becomes active. The link-protecting post-convergence path is R1-R21-R22-R23-R24-R3. The top label on the label stack is 801024 and corresponds to the node SID to reach R24. The 801003 label corresponds to the node SID on R23 to reach R3 on the shortest path R23-R2-R3.

## Verify adjacency SIDs

Verify adjacency SIDs of devices that have IS-IS adjacencies with Device R1.



**NOTE:** The SID values can vary in your configuration setup.

## Action

From operational mode, run the `show isis adjacency detail` command to display the adjacency information on Device R1.

```
user@R1> show isis adjacency detail
R21
  Interface: ge-0/0/0.0, Level: 1, State: Up, Expires in 19 secs
  Priority: 0, Up/Down transitions: 3, Last transition: 07:06:07 ago
  Circuit type: 1, Speaks: IP, IPv6
  Topologies: Unicast
  Restart capable: Yes, Adjacency advertisement: Advertise
  IP addresses: 10.10.20.2
  Level 1 IPv4 Adj-SID: 299840

R31
  Interface: ge-0/0/1.0, Level: 1, State: Up, Expires in 22 secs
  Priority: 0, Up/Down transitions: 3, Last transition: 07:06:07 ago
  Circuit type: 1, Speaks: IP, IPv6
  Topologies: Unicast
  Restart capable: Yes, Adjacency advertisement: Advertise
  IP addresses: 10.10.30.2
  Level 1 IPv4 Adj-SID: 299808

R2
  Interface: ge-0/0/2.0, Level: 1, State: Up, Expires in 24 secs
```

```

Priority: 0, Up/Down transitions: 3, Last transition: 07:06:07 ago
Circuit type: 1, Speaks: IP, IPv6
Topologies: Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 10.10.10.2
Level 1 IPv4 Adj-SID: 299776

```

## Meaning

Adjacency SIDs are assigned to each adjacency of Device R1 in the segment routing domain:

- Device R21 - 299840
- Device R31 - 299808
- Device R2 - 299776

The adjacency SIDs have local significance and can be used to steer traffic along specific outgoing interfaces. When you do not configure adjacency SIDs, they are dynamically assigned with a value outside of the default (or configured) SRGB range.

### *Verify the TI-LFA routes using adjacency SIDs*

## Purpose

Increase the cost of the post-convergence path from R1 to R3 and verify the TI-LFA routes using adjacency SIDs to avoid the primary path to reach the destination, Device R3.

## Action

From the configuration mode, increase the cost of the interface connecting Device R22 and R23, ge-0/0/0.

```

[edit protocols]
user@R22# set protocols isis interface ge-0/0/0.0 level 1 metric 1000
user@R22# commit

```

From operational mode, again run the `show route 192.168.0.3` command.

```

user@R1> show route 192.168.0.3

```

```

inet.0: 38 destinations, 38 routes (38 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.0.3/32    *[IS-IS/15] 10:44:56, metric 20
                  > to 10.10.10.2 via ge-0/0/2.0

inet.3: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.0.3/32    *[L-ISIS/14] 00:00:31, metric 20
                  > to 10.10.10.2 via ge-0/0/2.0, Push 801003
                  to 10.10.30.2 via ge-0/0/1.0, Push 801003, Push 299808(top)

```

## Meaning

The TI-LFA backup paths are now using the adjacency SID (in this case, 299808) instead of the node SID (801003) to reach Device R3. This is because node SIDs always use the shortest path between two nodes, and when the R22-R23 link cost went up, the shortest path to R1 overlaps with the primary path. Because TI-LFA cannot take a primary path to reach the destination, adjacency SIDs are used to take R31-R34 as the new post-convergence path to reach Device R3.

## RELATED DOCUMENTATION

[Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS and OSPF | 524](#)

[Configuring Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS | 529](#)

*post-convergence-lfa*

*use-post-convergence-lfa*

*use-for-post-convergence-lfa*

*node-protection*

[vLab Sandbox: Segment Routing - Basic](#)

## Static Adjacency Segment Identifier for IS-IS and OSPF

Adjacency segment is a strict forwarded single-hop tunnel that carries packets over a specific link between two nodes, irrespective of the link cost. You can configure static adjacency segment identifier (SID) labels for an interface or an interface group.

Configuring a static adjacency SID on an interface causes the existing dynamically allocated adjacency SID to be removed along with the transit route for the same.

For static adjacency SIDs, the labels are picked from either a static reserved label pool or from the segment routing global block (SRGB).

You can reserve a label range to be used for static allocation of labels using the following configuration:

```
user@host# set protocols mpls label-range static-label-range start-value end-value
```

The static pool can be used by any protocol to allocate a label in this range. You need to ensure that no two protocols use the same static label. The adjacency SIDs can be allocated from this label block through the configuration using keyword `label`. The label value for the specific adjacency SIDs need to be explicitly configured. The specific label is advertised as the adjacency SIDs for that interface for the specific level and address family.

The following is a sample configuration for IS-IS:

```
user@host# set protocols mpls label-range static-label-range 700000 799999;
user@host# set protocols isis source-packet-routing srgb start-label 800000 index-range 4000;
user@host# set protocols isis interface ge-0/0/0.1 level 1 ipv4-adjacency-segment unprotected label 700001;
```

The following is a sample configuration for OSPF:

```
user@host# set protocols mpls label-range static-label-range 700000 799999;
user@host# set protocols ospf source-packet-routing srgb start-label 800000 index-range 4000;
user@host# set protocols ospf area 0.0.0.0 interface ge-0/0/0.1 ipv4-adjacency-segment unprotected label 700001;
```



**NOTE:** When you use `ipv4-adjacency-segment` command, the underlying interface must be point-to-point.

SRGB is a global label space that is allocated for the protocol based on configuration. The labels in the entire SRGB is available for ISIS to use and are not allocated to other applications/protocols. Prefix SIDs (and Node SIDs) are indexed from this SRGB.

The Adj-SIDs can be allocated from the SRGB using keyword 'index' in the configuration. In such cases, it should be ensured that the Adj-SID index does not conflict with any other prefix SID in the domain. Like Prefix-SIDs, Adj-SIDs will also be configured by mentioning the index with respect to the SRGB. However, the Adj-SID subtlv will still have the SID as a value and the L and V flags are set.

The following is a sample configuration for IS-IS:

```
user@host# set protocols isis source-packet-routing srgb start-label 800000 index-range 4000;
user@host# set protocols isis interface ge-0/0/0.1 level 1 ipv4-adjacency-segment unprotected index 1;
```

Static adjacency SIDs can be configured per address family and also based on whether the protection is required or not. Adjacency SIDs should be configured per level per interface at the [edit protocols isis interface *interface-name* level *level-num*] hierarchy level.

The following is a sample configuration for OSPF:

```
user@host# set protocols ospf source-packet-routing srgb start-label 800000 index-range 4000;
user@host# set protocols ospf area 0.0.0.0 interface ge-0/0/0.1 ipv4-adjacency-segment unprotected index 1;
```

Static adjacency SIDs can be configured per area and also based on whether the protection is required or not. Adjacency SIDs should be configured per interface at the [edit protocols ospf area *area* interface *interface-name*] hierarchy level.

- Protected—Ensures adjacency SID is eligible to have a backup path and a B-flag is set in an adjacency SID advertisement.
- Unprotected—Ensures no backup path is calculated for a specific adjacency SID and a B-flag is not set in an adjacency SID advertisement.

The following is a sample configuration for IS-IS:

```
user@host# set protocols isis interface ge-0/0/0.1 level 1 ipv4-adjacency-segment unprotected index 1;
user@host# set protocols isis interface ge-0/0/1.1 level 1 ipv4-adjacency-segment protected index 2;
```

The following is a sample configuration for OSPF:

```
user@host# set protocols ospf area 0.0.0.0 interface ge-0/0/0.1 ipv4-adjacency-segment unprotected index 1;
user@host# set protocols ospf area 0.0.0.0 interface ge-0/0/1.1 ipv4-adjacency-segment protected index 2;
```

You can use the same adjacent SID for multiple interfaces by grouping a set of interfaces under an interface group and configuring the adjacency SID for that interface group and traffic can be load balanced among the interfaces under the interface group using weight. This can be configured under the [edit protocols isis interface-group *interface\_group\_name*] hierarchy level.

When segment routing is used in LAN subnetworks, each router in the LAN may advertise the adjacency SID of each of its neighbors. To configure adjacency SID for a LAN interface to a specific neighbor, configure the adjacency SIDs under the lan-neighbor configuration at the [edit protocols isis interface

*interface\_name* level *level\_num* lan-neighbor *neighbor-sysid*] hierarchy level for IS-IS and [[edit protocols ospf area 0.0.0.0 interface *interface\_name* lan-neighbor *neighbor-routerid*]] hierarchy level for OSPF. The following is a sample configuration for IS-IS:

```
user@host# set protocols mpls label-range static-label-range 700000 799999;
user@host# set protocols isis source-packet-routing srgb start-label 800000 index-range 4000;
user@host# set protocols isis interface ge-0/0/0.1 level 1 lan-neighbor 1234.1234.1234 ipv4-adjacency-
segment unprotected label 700001;
```

The following is a sample configuration for OSPF:

```
user@host# set protocols mpls label-range static-label-range 700000 799999;
user@host# set protocols ospf source-packet-routing srgb start-label 800000 index-range 4000;
user@host# set protocols ospf area 0.0.0.0 interface ge-1/0/0.1 lan-neighbor 11.12.1.2 ipv4-adjacency-
segment unprotected label 700001;
```

An adjacency set can be configured by declaring a set of interfaces under an interface group and configuring the adjacency segment for that interface group. The adjacency SID can be picked from the reserved static label pool or ISIS SRGB. Unlike normal interfaces, dynamic adjacency SID is not allocated by default under interface group, in which case the `dynamic` CLI statement is configured. Interfaces configured under an interface group can also be configured separately as independent interfaces as long as the `link-group-protection` is not configured. The following is a sample configuration:

```
user@host# set protocols mpls label-range static-label-range 700000 799999;
user@host# set protocols isis source-packet-routing srgb start-label 800000 index-range 4000;
user@host# set protocols isis interface-group group1 interface ge-0/0/0.1 weight 1;
user@host# set protocols isis interface-group group1 interface ge-0/0/1.1 weight 2;
user@host# set protocols isis interface-group group1 ipv4-adjacency-segment unprotected label 700001;
```

Use the following CLI hierarchy for configuring adjacency SID for IS-IS:

```
[edit ]
protocols {
  isis {
    interface <interface_name> {
      level <level_num> {
        ipv4-adjacency-segment {
          protected {
            dynamic;
            label <value>
```

```

        index <index>
    }
    unprotected {
        dynamic;
        label <value>
        index <index>
    }
}
ipv6-adjacency-segment {
    protected {
        dynamic;
        label <value>
        index <index>
    }
    unprotected {
        dynamic;
        label <value>
        index <index>
    }
}
}
interface <interface_name> {
    level <level_num> {
        lan-neighbor <neighbor-sysid>{
            ipv4-adjacency-segment {
                protected {
                    dynamic;
                    label <value>
                    index <index>
                }
                unprotected {
                    dynamic;
                    label <value>
                    index <index>
                }
            }
            ipv6-adjacency-segment {
                protected {
                    dynamic;
                    label <value>
                    index <index>
                }
            }
        }
    }
}

```





Use the following CLI hierarchy for configuring adjacency SID for OSPF:

```
[edit ]
protocols {
  ospf {
    area 0.0.0.0 {
      interface <interface_name> {
        ipv4-adjacency-segment {
          protected {
            dynamic;
            label <value>
            index <index>
          }
          unprotected {
            dynamic;
            label <value>
            index <index>
          }
        }
      }
    }
    interface <interface_name> {
      lan-neighbor <neighbor-routerid>{
        ipv4-adjacency-segment {
          protected {
            dynamic;
            label <value>
            index <index>
          }
          unprotected {
            dynamic;
            label <value>
            index <index>
          }
        }
      }
    }
  }
}
```

Use the following operational CLI commands to verify the configuration:

**show isis adjacency detail**

The following sample output displays the details of configured and dynamic adjacency SID.

```
user@host> show isis adjacency r1 detail
r1
  Interface: ge-0/0/2.1, Level: 1, State: Up, Expires in 19 secs
  Priority: 64, Up/Down transitions: 1, Last transition: 01:23:38 ago
  Circuit type: 3, Speaks: IP, IPv6, MAC address: 0:5:86:48:49:0
  Topologies: Unicast
  Restart capable: Yes, Adjacency advertisement: Advertise
  LAN id: r0.03, IP addresses: 11.1.1.2
  IPv6 addresses: fe80::205:8600:148:4900
  Level 1 IPv4   protected Adj-SID: 4138, Flags: BVL
  Level 1 IPv6 unprotected Adj-SID: 4139, Flags: FVL
```

**show ospf neighbor detail**

The following sample output displays the details of configured and dynamic adjacency SID.

```
user@host> show ospf neighbor detail
Address          Interface          State    ID              Pri  Dead
11.12.1.2        ge-1/0/0.0         Full     12.1.1.1        128   34
  Area 0.0.0.0, opt 0x52, DR 0.0.0.0, BDR 0.0.0.0
  Up 00:06:27, adjacent 00:06:27
  SPRING Adjacency Labels:

    Label      Flags      Adj-Sid-Type
    90010      BVLP      Protected
    1212       VLP       UnProtected
regress@10.49.129.231# run show route label 90010

mpls.0: 19 destinations, 19 routes (19 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

90010          *[L-OSPF/10/5] 00:00:21, metric 0
                > to 11.12.1.2 via ge-1/0/0.0, Pop
                  to 11.12.2.2 via ge-1/0/2.0, Swap 16021
                  to 11.12.3.2 via ge-1/0/3.0, Swap 16021
```

**show isis database extensive**

The following sample output displays the details of LAN/PTP adjacency SID.

```

user@host> show isis database extensive

r0.00-00 Sequence: 0x16, Checksum: 0xf156, Lifetime: 960 secs
  IPV4 Index: 1000, IPV6 Index: 2000
  Node Segment Blocks Advertised:
    Start Index : 0, Size : 4096, Label-Range: [ 16, 4111 ]
    IS neighbor: r4.00                      Metric:      10
      Two-way fragment: r4.00-00, Two-way first fragment: r4.00-00
    IS neighbor: r0.03                      Metric:      10
      Two-way fragment: r0.03-00, Two-way first fragment: r0.03-00
    IP prefix: 10.10.10.10/32                Metric:      0 Internal Up
    IP prefix: 11.1.1.0/24                  Metric:     10 Internal Up
    IP prefix: 21.1.1.0/24                  Metric:     10 Internal Up
    V6 prefix: 1001::/64                    Metric:     10 Internal Up
    V6 prefix: 2001::/64                    Metric:     10 Internal Up
    V6 prefix: abcd::10:10:10:10/128        Metric:      0 Internal Up

...
  TLVs:
    Area address: 49.00 (2)
    LSP Buffer Size: 1492
    Speaks: IP
    Speaks: IPV6
    IP router id: 10.10.10.10
    IP address: 10.10.10.10
    Hostname: r0
    IS neighbor: r0.03, Internal, Metric: default 10
    IS neighbor: r4.00, Internal, Metric: default 10
    IS extended neighbor: r0.03, Metric: default 10
    IP address: 11.1.1.1
    Local interface index: 342, Remote interface index: 0
    Current reservable bandwidth:
      Priority 0 : 1000Mbps
      Priority 1 : 1000Mbps
      Priority 2 : 1000Mbps
      Priority 3 : 1000Mbps
      Priority 4 : 1000Mbps
      Priority 5 : 1000Mbps

```

```

    Priority 6 : 1000Mbps
    Priority 7 : 1000Mbps
Maximum reservable bandwidth: 1000Mbps
Maximum bandwidth: 1000Mbps
    Administrative groups:  0 <none>
    LAN IPV4 Adj-SID: 4138, Weight:0, Neighbor:r1, Flags: BVL
    LAN IPV6 Adj-SID: 4139, Weight:0, Neighbor:r1, Flags: FBVL
IS extended neighbor: r4.00, Metric: default 10
IP address: 21.1.1.1
Neighbor's IP address: 21.1.1.2
Local interface index: 334, Remote interface index: 335
Current reservable bandwidth:
    Priority 0 : 1000Mbps
    Priority 1 : 1000Mbps
    Priority 2 : 1000Mbps
    Priority 3 : 1000Mbps
    Priority 4 : 1000Mbps
    Priority 5 : 1000Mbps
    Priority 6 : 1000Mbps
    Priority 7 : 1000Mbps
Maximum reservable bandwidth: 1000Mbps
Maximum bandwidth: 1000Mbps
Administrative groups:  0 <none>
P2P IPV4 Adj-SID - Flags: BVL,  Weight:0, Label: 4125
P2P IPV6 Adj-SID - Flags: FBVL, Weight:0, Label: 4126

```

### **show isis interface-group**

The following sample output displays the status information about the specified interface group.

```

user@host> show isis interface-group
Interface-group: r1r2ig
  ge-0/0/1.1, 1000Mbps, Up, Non-Degraded, Weight: 1
  ge-0/0/1.3, 1000Mbps, Up, Non-Degraded, Weight: 1
  ge-0/0/1.5, 1000Mbps, Up, Non-Degraded, Weight: 1
Total Nominal Bandwidth: 3Gbps, Total Actual Bandwidth: 3Gbps
Level 1 IPv4  protected Adj-SID: Label 4138
Level 1 IPv6 unprotected Adj-SID: Label 4139

```

## Overview of Segment Routing over RSVP Forwarding Adjacency in IS-IS

Segment routing architecture enables the ingress nodes in a core network to steer traffic through explicit paths through the network. The architecture provides the mechanism to enable source routing. Paths are encoded as sequences of topological subpaths called segments, which are advertised by link-state routing protocols such as IS-IS and OSPF.

A *forwarding adjacency* is a traffic engineered label-switched path (LSP) that is configured between two nodes and that is used by the interior gateway protocol (IGP) to forward traffic. The forwarding adjacency creates a tunneled path for sending data between peer devices in an RSVP LSP network.

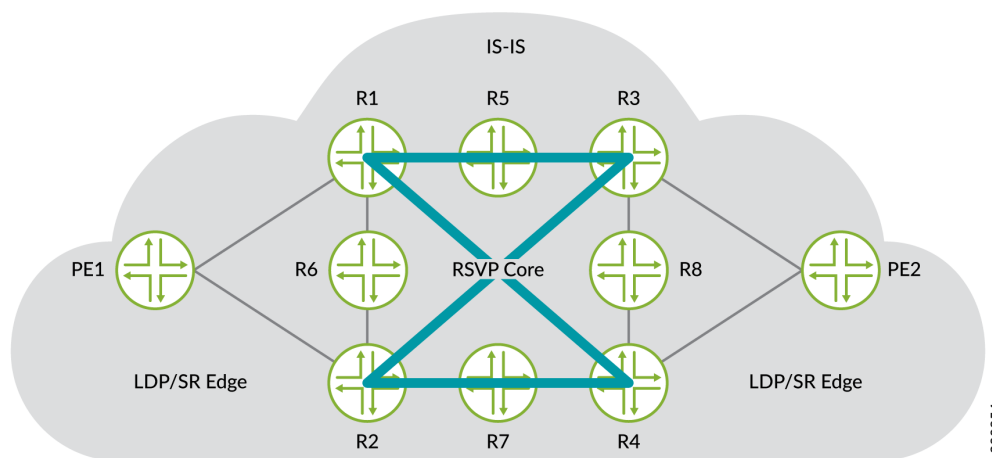
Junos OS supports segment routing traffic to be carried over RSVP LSPs that are advertised as forwarding adjacencies in IS-IS.

### Benefits of Segment Routing over RSVP LSPs

- Reduces network complexity by removing protocols such as LDP.
- Leverages IGPs such as IS-IS, and RSVP for efficient and flexible forwarding.
- Provides a faster and more efficient way of forwarding traffic in the RSVP core network.

Figure 43 on page 559 illustrates the typical deployment network for segment routing over RSVP forwarding adjacency.

**Figure 43: Segment Routing over RSVP Forwarding Adjacency**



The network consists of provider edge (PE) routers configured with LDP on the edge and RSVP in the core. You can easily replace LDP with IS-IS segment routing because segment routing eliminates the

need for MPLS signaling protocols such as LDP. As a result, you enable network simplification by removing a protocol from the network.

### How IS-IS Segment Routing over RSVP Forwarding Adjacency Works

RSVP LSPs are configured as links in IS-IS. IS-IS builds dummy adjacencies over these links (no hellos) and advertises them as links in LSPs. Because RSVP LSPs are advertised as forwarding adjacencies, the LDP or segment routing edge nodes can forward traffic towards appropriate core nodes. The metric on RSVP LSPs is manipulated to manage traffic steering from the head node to the end nodes. RSVP uses the shortest-path-first (SPF) algorithm to compute the shortest path to all nodes in the network. As a result, when IP routes point to RSVP LSPs, segment routing routes also point to these LSPs. This is because segment routing reuses the SPF computation performed for the IP routes.

### RELATED DOCUMENTATION

*Understanding Source Packet Routing in Networking (SPRING)*

## IGP Microloop Avoidance

### SUMMARY

Microloops can consume the available bandwidth of the links, which impacts the efficient transmission of useful packets. Microloop avoidance can prevent forwarding of looping packets.

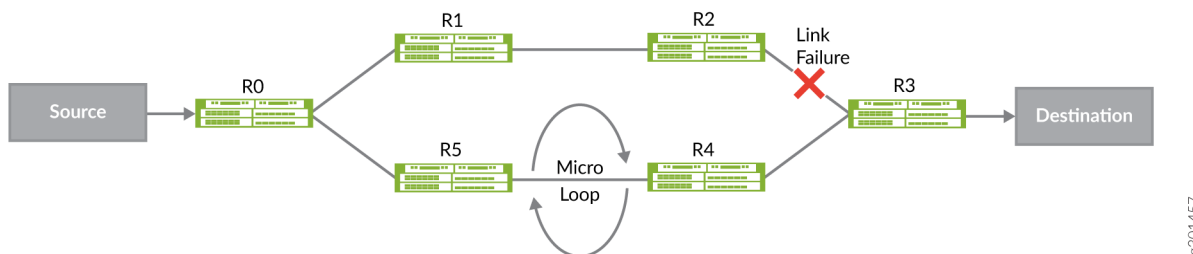
### IN THIS SECTION

- [Benefits of Avoiding Microloops | 560](#)
- [Microloop Avoidance in SR-MPLS Networks | 561](#)
- [Supported Platforms and Unsupported Features | 563](#)

### Benefits of Avoiding Microloops

- Micro loop-free path avoids delays and traffic loss
- Microloop avoidance can prevent forwarding of looping packets and avoid wasteful bandwidth consumption
- Microloop avoidance path is computed only for the impacted links in case of multiple link failures. If the second link failure does not impact the computed microloop avoidance path, IGP continues to use the same microloop avoidance path.

Junos OS enables a device to defer IS-IS route download when an IS-IS link fails in order to avoid micro loops. When local links go down, the IS-IS protocol floods an entire area with the database. If the node connected to the local interface that has failed converges faster than the neighboring node, then the connected node redirects traffic to the converged path. This redirection can result in micro looping of traffic until the neighboring node converges. When the primary path of a protected node fails, the connected node does not need to converge quickly if the configured backup path is not impacted. In this case, traffic flow towards a converged path is deferred until the configured delay time. This time delay helps in avoiding microloops because all routers do not arrive at the post-convergence forwarding states simultaneously.



In the "figure" on page 560, the primary path from Source to Destination is S→R0→R1→R2→R3→D. When the link between R2 and R3 fails, traffic sent from S to D, is subject to transient forwarding loops while routers update their forwarding state for destination D.

- If R0 updates its forwarding state before R5, packets will loop between R0 and R5
- If both R0 and R5, have updated their forwarding states, and R4 has not, packets will loop between R4 and R5.
- R0 detects the link failure between R2 and R3, and temporarily steers traffic destined to Destination over SR path [NodeSID(R4), AdjSID(R4→R3), D].
- When the configured timeout elapses, R0 just uses the node-SID to D to reach the destination.

## Microloop Avoidance in SR-MPLS Networks

Enable post-convergence path calculation on a device to avoid microloops between network devices. Microloops form when a network change such as a link or metric change occurs in a segment routing MPLS network. A network change might trigger a loop between upstream and downstream routers for a brief time period because the routers do not update their forwarding state simultaneously.

To configure microloop avoidance in a segment routing MPLS network, include the `maximum-labels` and the `maximum-srv6-sids` statements at the `[edit protocols isis spf-options microloop-avoidance post-convergence-path]` hierarchy level.

When an IPV6 prefix has both SR-MPLS- MLA and SRV6 micro-loop-avoiding paths available, we will prefer the SR-MPLS MLA path. SR-MPLS can provide micro-loop-avoiding paths for ipv4/ipv6 prefixes and SR-labels. Delay specifies the time in milliseconds for which we use the Micro-loop-Avoidance path, before transitioning to SPF path. Note that microloop avoidance is not a replacement for local repair mechanisms such as topology-independent loop-free alternate (TI-LFA), which detects local

failure very fast and activates a precomputed loop-free alternative path. Routers that implement micro-loop avoidance compute the micro-loop avoiding path only after receiving the link state update for the event. So, micro-loop avoidance mechanism is not a replacement for local repair mechanisms like TI-LFA which detect local failure very fast and activate a pre-computed loop-free-alternative path at PFE level. In the above example, if local repair mechanism is not present for the R2-R3 failure, there will be lot of traffic loss before R0 can detect the failure(via global convergence) and program a micro-loop avoiding path. Micro-loop avoidance can't avoid traffic loss due to delayed detection of the failure. Micro-loop avoidance will avoid traffic loss due to micro-loops only. Both local-repair mechanisms like TI-LFA and micro-loop avoidance, will have to be enabled on all the nodes in the network to ensure that traffic loss is in milli-seconds range.

To avoid micro-loops, the following process is used:

1. After computing the new path to D, for a predetermined time, R installs an entry for D that steers packets to D via a loop-free SR path. This time should be greater than worst case delay of any router in the network.
2. After the configured time delay, R installs the post-convergence route entry for D, which is without any SIDs.



**NOTE:** If microloop avoidance is configured for both SRv6 and SR-MPLS, IS-IS prefers to take the SR-MPLS path.

In case the microloop path requires more than the number of SIDs (SR-MPLS or SRv6 SIDs) the platform supports, then only the global convergence primary path is installed.

To configure microloop avoidance in an OSPFv2 segment routing network for both local and remote network events including link down, link-up, and metric-change, configure:

- The maximum number of labels installed for post-convergence paths by including the `maximum-labels` statement at the `[edit protocols ospf spf-options microloop avoidance post-convergence-path]` hierarchy level.
- The time after which temporary post-convergence paths are removed by including the `delay milliseconds` statement at the `[edit protocols ospf spf-options microloop avoidance post-convergence-path]` hierarchy level.

For effective microloop avoidance, configure this feature on all the nodes in the network.



**NOTE:** For OSPFv2, Microloop avoidance is supported for IPv4 networks only.



## Supported Platforms and Unsupported Features

Junos OS supports microloop avoidance on most platforms that support IS-IS. For details on specific devices and Junos OS releases that support IS-IS micro loop avoidance, see [Feature Explorer](#).

Junos OS does not support the following features in conjunction with microloop avoidance:

- Cannot prevent traffic loss because of slow control plane convergence.
- If shortcuts are available IGP does not provide a microloop avoidance path.
- Microloop avoidance path that needs more than 8 labels is not supported for OSPFv2. The maximum number of labels installed for microloop avoidance path is 8. For the microloop avoidance ECMP path to be usable, the number of labels must be less than or equal to maximum labels.
- Adjacency SIDs are not supported with OSPFv2 microloop avoidance.
- OSPFv2 multi-topology is not supported with microloop avoidance.

### RELATED DOCUMENTATION

| *microloop-avoidance*

## Overview of SRv6 Network Programming in IS-IS Networks

### IN THIS SECTION

- [Benefits of SRv6 Network Programming | 564](#)
- [SRv6 Network Programming Overview | 564](#)
- [What Is a Segment Routing Extension Header? | 565](#)
- [Flexible Algorithm for SRv6 Dataplane | 566](#)
- [TI-LFA for SRv6 | 567](#)
- [NSR Support for SRv6 | 568](#)
- [Supported and Unsupported Features for SRv6 Network Programming in IS-IS | 568](#)

## Benefits of SRv6 Network Programming

SRv6 network programming provides the following benefits in an IPv6 network:

- **Seamless deployment:** Network programming depends entirely on the IPv6 header and the header extension to transport a packet, eliminating protocols such as MPLS. This eases deployment without the need for any major hardware or software upgrade in a core IPv6 network.
- **Flexible deployment:** An SRv6 ingress node can transport packets even if transit routers aren't SRv6-capable. This eliminates the need to deploy segment routing on all nodes in an IPv6 network.
- **Single-device versatility:** Junos OS supports multiple functions on a single segment identifier (SID) and can inter-operate in the insert mode and the encapsulation mode. This allows a single device to simultaneously perform the provider (P) router and the provider edge (PE) router roles.

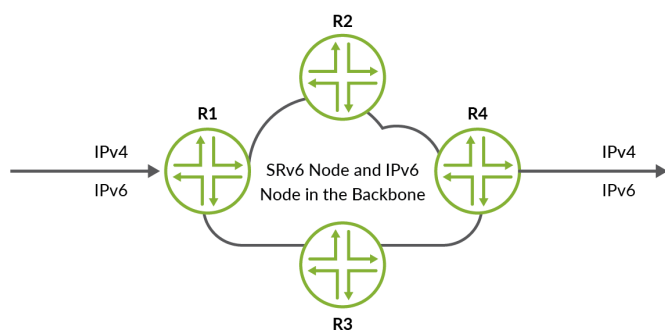
## SRv6 Network Programming Overview

Network programming is the capability of a network to encode a network program into individual network instructions that are then inserted into the IPv6 packet headers. The IPv6 packet carrying the network instructions explicitly tells the network about the precise SRv6 nodes available for packet processing. The network instruction is the SRv6 SID that is represented by a 128-bit IPv6 address. The SIDs are distributed through the network in the IPv6 packet headers. Along with the addressing, network instructions define a particular task or function for each SRv6-capable node in the SRv6 network.



**NOTE:** You can configure segment routing in a core IPv6 network without an MPLS data plane on MX Series devices with MPC7E, MPC8E and MPC9E line cards.

This feature is useful for service providers whose networks are predominantly IPv6 and have not deployed MPLS. Such networks depend only on IPv6 headers and header extensions for transmitting data. This feature also benefits networks that need to deploy segment routing traffic through transit routers that do not have segment routing capability yet. In such networks, the SRv6 network programming feature can provide flexibility to leverage segment routing without deploying MPLS.



g301239

## What Is a Segment Routing Extension Header?

A Segment Identifier represents a specific segment in a segment routing domain. In an IPv6 network, the SID-type used is a 128-bit IPv6 address also referred to as an SRv6 Segment or SRv6 SID. SRv6 stacks up these IPv6 addresses instead of MPLS labels in a segment routing extension header. The Segment Routing Extension Header (SRH) is a type of IPv6 routing extension header. Typically, the SRH contains a segment list encoded as an SRv6 SID. An SRv6 SID consists of the following parts:

- **Locator**—Locator is the first part of an SID that consists of the most significant bits representing the address of a particular SRv6 node. The locator is very similar to a network address that provides a route to its parent node. The IS-IS protocol installs the locator route in the `inet6.0` routing table. IS-IS routes the segment to its parent node, which subsequently performs a function defined in the other part of the SRv6 SID. You can also specify the algorithm associated with this locator. You can define a flexible algorithm as per your network requirements.
- **Function**—The other part of the SID defines a function that is performed locally on the node that is specified by the locator. There are several functions that have already been defined in RFC 8986 *Segment Routing over IPv6 (SRv6) Network Programming*. However, the following functions that are signalled in IS-IS are available on Junos OS. . IS-IS installs these function SIDs in the `inet6.0` routing table.
  - **End**—An endpoint function for an SRv6 instantiation of a Prefix SID. It does not allow for decapsulation of an outer header for the removal of an SRH. Therefore, an End SID cannot be the last SID of a SID list and cannot be the Destination Address (DA) of a packet without an SRH (unless combined with the PSP, USP or USD flavors).
  - **End.X**—An endpoint X function is an SRv6 instantiation of an adjacent SID. It is a variant of the endpoint function with Layer 3 cross-connect to an array of Layer 3 adjacencies.

You can specify an End SID behavior such as Penultimate Segment Pop (PSP), Ultimate Segment Pop (USP), or Ultimate Segment Decapsulation (USD).

- **PSP**—When the last SID is written in the destination address, the End and End.X functions with the PSP flavor pop the top-most SRH. Subsequent stacked SRHs may be present but are not processed as part of the function.
- **USP**—When the next header is an SRH and there are no more segments left, the IS-IS protocol pops the top SRH, looks up the updated destination address and forwards the packet based on match table entry.
- **USD**—When the next header in the packet is 41 or is an SRH and there are no more segments left, then IS-IS pops the outer IPv6 header and its extension headers, looks up the exposed inner IP destination address and forwards the packet to the matched table entry.



**NOTE:** The size of the locator and function is flexible, and you can customize the size as per your requirements. You must configure the locator before you define the functions. Each locator can advertise multiple end SIDs and end.X SIDs that are associated with it. Ensure that the locator and SIDs belong to the same subnet to avoid commit error.

For example, you can have an SRv6 SID: 2001:DB8:AC05:FF01:A000:: where 2001:DB8:AC05:FF01 represents 64-bit locator and A000 represents 16-bit function:

**Table 7: 128-bit SRv6 SID**

Locator	Function
2001:db8:AC05:FF01	A000

### Flexible Algorithm for SRv6 Dataplane

In a core IPv6 domain configured with segment routing you can define flexible algorithms that compute paths using different parameters and link constraints based on your requirements. For example, you can define a flexible algorithm that computes a path to minimize the IGP metric and define another flexible algorithm to compute a path based on the traffic engineering metrics to divide the network into separate planes. You can configure the flexible algorithm locators to steer packets along the constraint-based paths in an SRv6 domain.

To configure a flexible algorithm for SRv6, see [Flexible Algorithms in IS-IS for Segment Routing Traffic Engineering](#)

To advertise the flexible algorithm mapped to the locator, include the `algorithm` option at the `[edit protocols isis source-packet-routing srv6 locator]` hierarchy level. The mapped flexible algorithm is applied to End SIDs and End-X-SIDs under SRv6 locators.



**NOTE:** If a node is participating in a specific flexible algorithm it applies to both SR MPLS and SRv6 nodes. You cannot define flexible algorithms specifically for either SR MPLS or SRv6.

For ingress traffic, Junos OS uses the encapsulation mode by default. Therefore the destination needs to have USD capable SIDs. Other SRH anchor nodes in the flexible algorithm path can be of any flavor.

For transit traffic in the insert mode, the last anchor node for the flexible algorithm path must have a PSP-capable SID. In the absence of the PSP-capable SID, IS-IS does not download a path through that anchor node. In such cases, IS-IS downloads other ECMP paths with the appropriate flavored SIDs.

## TI-LFA for SRv6

Topology Independent- Loop Free Alternate (TI-LFA) establishes a Fast Reroute (FRR) path that is aligned to a post-convergence path. An SRv6-capable node inserts a single segment into the IPv6 header or multiple segments into the SRH. Multiple SRHs can significantly raise the encapsulation overhead, which can sometimes be more than the actual packet payload. Therefore, by default, Junos OS supports SRv6 tunnel encapsulation with reduced SRH. The point-of-local repair (PLR) adds the FRR path information to the SRH containing the SRv6 SIDs.

The TI-LFA backup path is represented as a group of SRv6 SIDs inside an SRH. At the ingress router, IS-IS encapsulates the SRH in a fresh IPv6 header. However, at transit routers, IS-IS inserts the SRH into the data traffic in the following manner:

- **Insert Mode**— IS-IS inserts an SRH as the next header in the original IPv6 packet header and modifies the next header according to the value of the SRH. The IPv6 destination address is replaced with the IPv6 address of the first SID in the segment list and the original IPv6 destination address is carried in the SRH header as the last segment in the list. To enable the insert mode at transit routers, include the `transit-srh-insert` statement at the `[edit protocols isis source-packet-routing srv6]` hierarchy level.
- **Encap Mode**— In the encap mode, the original IPv6 packet is encapsulated and transported as the inner packet of an IPv6-in-IPv6 encapsulated packet. The outer IPv6 packet carries the SRH with the segment list. The original IPv6 packet travels unmodified in the network. By default, Junos OS supports SRv6 tunnel encapsulation in reduced SRH. However, you can choose one of the following tunnel encapsulation methods:
  - **Reduced SRH (default)**— With the reduced SRH mode, if there is only one SID, there is no SRH added and the last SID is copied into the IPV6 destination address. You cannot preserve the entire SID list in the SRH with a reduced SRH.
  - **Non-reduced SRH**— You can configure the non-reduced SRH tunnel encapsulation mode when you want to preserve the entire SID list in the SRH.

To configure non-reduced SRH, include the `no-reduced-srh` statement at the `[edit routing-options source-packet-routing srv6]` hierarchy level.



**NOTE:** If you configure or delete `no-reduced-srh` on ACX7000 platforms, it restarts the Packet Forwarding Engine (PFE).



**NOTE:** Currently, IPv6-only networks do not support fate sharing. Also, SRv6 TI-LFA does not take Shared Risk Link Group (SRLG) into consideration when computing backup

paths. For more information on TI-LFA, see ["Understanding Topology-Independent Loop-Free Alternate with Segment Routing for IS-IS"](#) on page 524.

## NSR Support for SRv6

We support IS-IS nonstop active routing (NSR) for dynamic classic adjacency End-x SIDs. Junos OS allocates the same dynamic SID on both the active and backup Routing Engines (RE) after switch-over to ensure dynamically allocated SIDs on the primary RE are not repurposed. You can also use BGP NSR for dynamic DT SIDs. Note that Junos OS currently does not support NSR for classic dynamic End SIDs.

## Supported and Unsupported Features for SRv6 Network Programming in IS-IS

SRv6 network programming in IS-IS networks currently supports:

- Core IPv6 and dual stack. IPv4 and IPv6 transport is supported for dual stack.
- IPv4 and IPv6 payloads.

**Table 8: Platform-Specific SID Support**

Platform	Description	Number of SIDs Supported
ACX7000 family of routers	Maximum number of SIDs supported in SRH	12
	Maximum number of SIDs that can be popped (removed)	7
	Maximum number of SIDs that can be inserted	10
	Maximum number of SIDs supported for encapsulation	12
MX Series	Maximum number of SIDs supported in SRH	6
	Maximum number of SIDs that can be popped (removed)	7

Table 8: Platform-Specific SID Support (*Continued*)

Platform	Description	Number of SIDs Supported
	Maximum number of SIDs that can be inserted	5
	Maximum number of SIDs supported for encapsulation	6
	Maximum number of SIDs supported for End.D (decapsulation) operations	6
PTX10002-36QDD, PTX10008 (Express 5-enhanced mode)	Maximum number of SIDs that can be popped (removed)	7
	Maximum number of SIDs that can be inserted	7
	Maximum number of SIDs supported for encapsulation	7

SRv6 network programming in IS-IS Networks currently does not support:

- Anycast for locator prefix.
- Shared Risk Link Group (SRLG) when computing backup paths.
- Static SRv6 tunnel with segment lists.
- ICMP and ICMPv6 error handling.
- SR-TE policy configuration for SRv6 Tunnel.
- Conflict resolution for Flexible Algorithm locators. Multiple nodes sharing the same locator prefix with different algorithm values could result in unexpected routing behavior.
- Interface group for End-X-SID.
- Configuration of normal or extended admin-groups for IPv6 networks without MPLS. These features can only be configured at [edit protocols mpls] hierarchy level.

## RELATED DOCUMENTATION

[\*srv6\*](#)[\*locator\*](#)[\*flex-algorithm\*](#)[\*definition\*](#)

## Example: Configuring SRv6 Network Programming in IS-IS Networks

### IN THIS SECTION

- [Requirements | 570](#)
- [Overview | 570](#)
- [Configuration | 572](#)
- [Verification | 589](#)

This example shows how to configure SRv6 network programming in an IS-IS network. This feature is useful for service providers whose networks are predominantly IPv6 and have not deployed MPLS. Such networks depend only on the IPv6 headers and header extensions for transmitting data. SRv6 network programming provides flexibility to leverage segment routing without deploying MPLS.

### Requirements

This example uses the following hardware and software components:

- Eight MX Series routers with MPC7E, MPC8E, or MPC9E line cards
- Junos OS Release 20.3R1 or later

### Overview

#### IN THIS SECTION

- [Topology | 571](#)



You can configure SRv6 without MPLS in a core IPv6 network. This feature benefits networks that need to deploy SR traffic through transit routers that do not have segment routing capability yet.

## Topology

In [Figure 44 on page 572](#), Router R0 and Router R7 are ingress and egress routers that support IPv4 only devices CE1 and CE2. Routers R1, R2, R3, R4, R5, and R6 comprise an IPv6 only provider core network. All routers belong to the same autonomous system. IS-IS is the interior gateway protocol in the IPv6 core and is configured to support SRv6. In this example the Router R2 is configured as an IPv6 route reflector with IBGP peering sessions to both R0 and R7. No other routers speak BGP in this example.

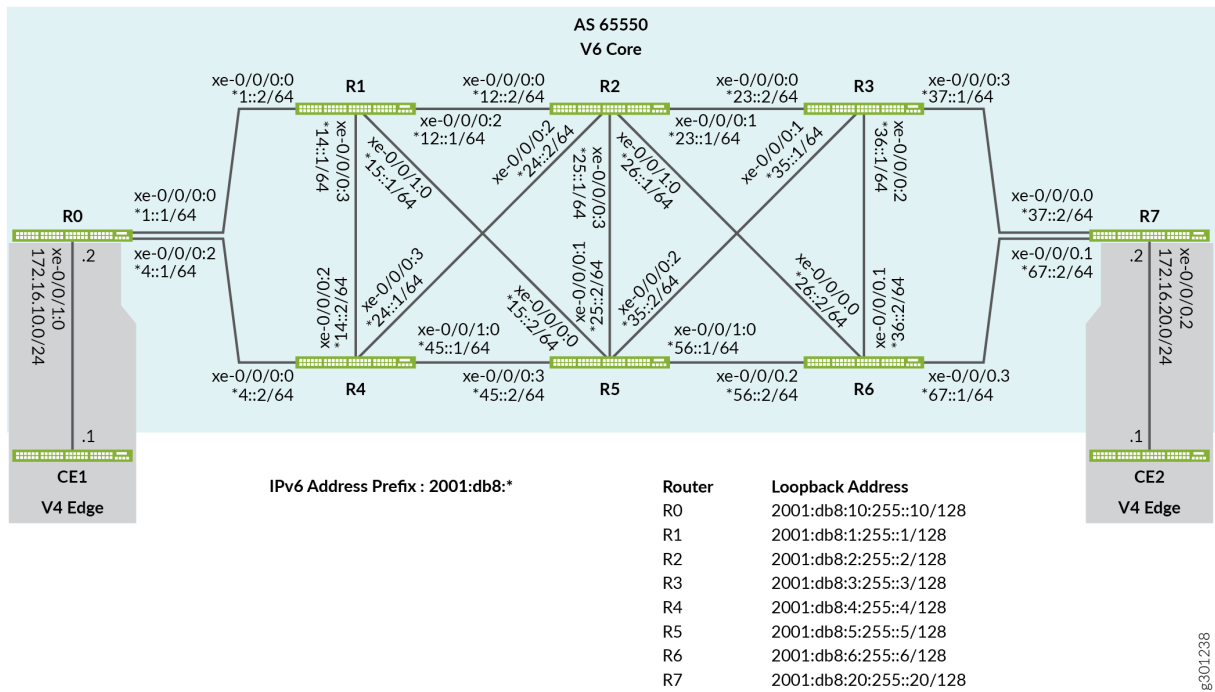


**NOTE:** To better demonstrate SRv6 tunneling, this example is based on a pure IPv6 provider core. SRv6 is supported with a dual stack core where both IPv6 and IPv4 are deployed.

The edge routers that support IPv4 devices need to transport IPv4 traffic using IPv6 tunnel encapsulation. The encapsulation tunnels are derived from SRv6 SIDs configured at SRv6-enabled routers. The IS-IS protocol processes these SRv6 SIDs and updates the inet6.3 table with the next-hop addresses of the available tunnel endpoints. When an IPv4 route is learned through BGP the router attempts to resolve the associated next hop through the inet6.3 table. When a matching entry is found the result is an automatic IPv6 tunnel to the endpoint that advertised the BGP route.

In this example both the R0 and R7 routers advertise their attached IPv4 subnet using BGP. This results in IPv6 tunnels between the edge routers. The tunnels are used to transport the IPv4 traffic over the IPv6 provider core. At egress, the edge routers decapsulate the outer IPv6 header and perform an IPv4 route lookup to forward the packet to its destination.

Figure 44: SRv6 Network Programming in IS-IS



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 572](#)
- [Configuring Router R0 | 582](#)
- [Results | 586](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter commit from configuration mode.

## Router R0

```

set interfaces xe-0/0/0:0 description To_R1
set interfaces xe-0/0/0:0 mtu 4000
set interfaces xe-0/0/0:0 unit 0 family iso
set interfaces xe-0/0/0:0 unit 0 family inet6 address 2001:db8:1::1/64
set interfaces xe-0/0/0:2 description To_R4
set interfaces xe-0/0/0:2 mtu 4000
set interfaces xe-0/0/0:2 unit 0 family iso
set interfaces xe-0/0/0:2 unit 0 family inet6 address 2001:db8:4::1/64
set interfaces xe-0/0/1:0 description To_CE1
set interfaces xe-0/0/1:0 unit 0 family inet address 172.16.10.2/24
set interfaces xe-0/0/1:0 unit 0 family iso
set interfaces lo0 unit 0 family iso address 49.0001.000a.0a0a.0a00
set interfaces lo0 unit 0 family inet6 address 2001:db8:10:255::10/128
set policy-options policy-statement pplb then load-balance per-packet
set routing-options source-packet-routing srv6 locator myloc 2001:db8:0:a0::/64
set routing-options forwarding-table export pplb
set routing-options router-id 172.16.255.10
set policy-options policy-statement CE1_v4 term 1 from protocol direct
set policy-options policy-statement CE1_v4 term 1 from route-filter 172.16.10.0/24 exact
set policy-options policy-statement CE1_v4 term 1 then next-hop 2001:db8:0:a0:d01::
set policy-options policy-statement CE1_v4 term 1 then accept
set routing-options autonomous-system 65550
set protocols bgp group to-R2RRv6 type internal
set protocols bgp group to-R2RRv6 export CE1_v4
set protocols bgp group to-R2RRv6 local-address 2001:db8:10:255::10
set protocols bgp group to-R2RRv6 neighbor 2001:db8:2:255::2 family inet unicast extended-nexthop
set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a0:1a01:: flavor psp
set protocols isis interface xe-0/0/0:0.0 node-link-protection
set protocols isis interface xe-0/0/0:0.0 point-to-point
set protocols isis interface xe-0/0/0:2.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a0:1a04:: flavor psp
set protocols isis interface xe-0/0/0:2.0 node-link-protection
set protocols isis interface xe-0/0/0:2.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srv6 locator myloc end-sid 2001:db8:0:a0:d01:: flavor
usd
set protocols isis level 1 disable

```

## Router R1

```

set interfaces xe-0/0/0:0 description To_R0
set interfaces xe-0/0/0:0 mtu 4000
set interfaces xe-0/0/0:0 unit 0 family iso
set interfaces xe-0/0/0:0 unit 0 family inet6 address 2001:db8:1::2/64
set interfaces xe-0/0/0:2 description To_R2
set interfaces xe-0/0/0:2 mtu 4000
set interfaces xe-0/0/0:2 unit 0 family iso
set interfaces xe-0/0/0:2 unit 0 family inet6 address 2001:db8:12:1/64
set interfaces xe-0/0/0:3 description to-R4
set interfaces xe-0/0/0:3 mtu 4000
set interfaces xe-0/0/0:3 unit 0 family iso
set interfaces xe-0/0/0:3 unit 0 family inet6 address 2001:db8:14::1/64
set interfaces xe-0/0/1:0 description to-R5
set interfaces xe-0/0/1:0 mtu 4000
set interfaces xe-0/0/1:0 unit 0 family iso
set interfaces xe-0/0/1:0 unit 0 family inet6 address 2001:db8:15::1/64
set interfaces lo0 unit 0 family iso address 49.0001.0001.0101.0100
set interfaces lo0 unit 0 family inet6 address 2001:db8:1:255::1/128
set policy-options policy-statement pplb then load-balance per-packet
set routing-options source-packet-routing srv6 locator myloc 2001:db8:0:a1::/64
set routing-options forwarding-table export pplb
set routing-options router-id 192.168.255.1
set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a1:1a10:: flavor psp
set protocols isis interface xe-0/0/0:0.0 node-link-protection
set protocols isis interface xe-0/0/0:0.0 point-to-point
set protocols isis interface xe-0/0/0:2.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a1:1a12:: flavor psp
set protocols isis interface xe-0/0/0:2.0 node-link-protection
set protocols isis interface xe-0/0/0:2.0 point-to-point
set protocols isis interface xe-0/0/0:2.1 node-link-protection
set protocols isis interface xe-0/0/0:3.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a1:1a14:: flavor psp
set protocols isis interface xe-0/0/0:3.0 node-link-protection
set protocols isis interface xe-0/0/0:3.0 point-to-point
set protocols isis interface xe-0/0/1:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a1:1a15:: flavor psp
set protocols isis interface xe-0/0/1:0.0 node-link-protection
set protocols isis interface xe-0/0/1:0.0 point-to-point
set protocols isis interface lo0.0 passive

```

```

set protocols isis source-packet-routing srv6 locator myloc end-sid 2001:db8:0:a1:d11:: flavor
usd
set protocols isis level 1 disable

```

## Router R2

```

set interfaces xe-0/0/0:0 description To_R1
set interfaces xe-0/0/0:0 mtu 4000
set interfaces xe-0/0/0:0 unit 0 family iso
set interfaces xe-0/0/0:0 unit 0 family inet6 address 2001:db8:12::2/64
set interfaces xe-0/0/0:1 description To_R3
set interfaces xe-0/0/0:1 mtu 4000
set interfaces xe-0/0/0:1 unit 0 family iso
set interfaces xe-0/0/0:1 unit 0 family inet6 address 2001:db8:23::1/64
set interfaces xe-0/0/0:2 description To_R4
set interfaces xe-0/0/0:2 mtu 4000
set interfaces xe-0/0/0:2 unit 0 family iso
set interfaces xe-0/0/0:2 unit 0 family inet6 address 2011:db8:24::1/64
set interfaces xe-0/0/0:3 description To_R5
set interfaces xe-0/0/0:3 mtu 4000
set interfaces xe-0/0/0:3 unit 0 family iso
set interfaces xe-0/0/0:3 unit 0 family inet6 address 2001:db8:25::1/64
set interfaces xe-0/0/1:0 description To_R6
set interfaces xe-0/0/1:0 mtu 4000
set interfaces xe-0/0/1:0 unit 0 family iso
set interfaces xe-0/0/1:0 unit 0 family inet6 address 2001:db8:26::1/64
set interfaces lo0 unit 0 family iso address 49.0001.0002.0202.0200
set interfaces lo0 unit 0 family inet6 address 2001:db8:2:255::2/128
set policy-options policy-statement pplb then load-balance per-packet
set routing-options source-packet-routing srv6 locator myloc 2001:db8:0:a2::/64
set routing-options forwarding-table export pplb
set routing-options router-id 192.168.255.2
set routing-options autonomous-system 65550
set protocols bgp group RRv6 type internal
set protocols bgp group RRv6 local-address 2001:db8:2:255::2
set protocols bgp group RRv6 neighbor 2001:db8:10:255::10 family inet unicast extended-next-hop
set protocols bgp group RRv6 neighbor 2001:db8:20:255::20 family inet unicast extended-next-hop
set protocols bgp group RRv6 cluster 192.168.255.2
set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a2:1a21:: flavor psp
set protocols isis interface xe-0/0/0:0.0 node-link-protection
set protocols isis interface xe-0/0/0:0.0 point-to-point

```

```

set protocols isis interface xe-0/0/0:1.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a2:1a23:: flavor psp
set protocols isis interface xe-0/0/0:1.0 node-link-protection
set protocols isis interface xe-0/0/0:1.0 point-to-point
set protocols isis interface xe-0/0/0:2.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a2:1a24:: flavor psp
set protocols isis interface xe-0/0/0:2.0 node-link-protection
set protocols isis interface xe-0/0/0:2.0 point-to-point
set protocols isis interface xe-0/0/0:3.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a2:1a25:: flavor psp
set protocols isis interface xe-0/0/0:3.0 node-link-protection
set protocols isis interface xe-0/0/0:3.0 point-to-point
set protocols isis interface xe-0/0/1:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a2:1a26:: flavor psp
set protocols isis interface xe-0/0/1:0.0 node-link-protection
set protocols isis interface xe-0/0/1:0.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srv6 locator myloc end-sid 2001:db8:0:a2:d21:: flavor
usd
set protocols isis level 1 disable

```

## Router R3

```

set interfaces xe-0/0/0:0 description To_R2
set interfaces xe-0/0/0:0 mtu 4000
set interfaces xe-0/0/0:0 unit 0 family iso
set interfaces xe-0/0/0:0 unit 0 family inet6 address 2001:db8:23::2/64
set interfaces xe-0/0/0:1 description To_R5
set interfaces xe-0/0/0:1 mtu 4000
set interfaces xe-0/0/0:1 unit 0 family iso
set interfaces xe-0/0/0:1 unit 0 family inet6 address 2001:db8:35::1/64
set interfaces xe-0/0/0:2 description To_R6
set interfaces xe-0/0/0:2 mtu 4000
set interfaces xe-0/0/0:2 unit 0 family iso
set interfaces xe-0/0/0:2 unit 0 family inet6 address 36::1/64
set interfaces xe-0/0/0:3 description To_R7
set interfaces xe-0/0/0:3 mtu 4000
set interfaces xe-0/0/0:3 unit 0 family iso
set interfaces xe-0/0/0:3 unit 0 family inet6 address 2001:db8:37::1/64
set interfaces lo0 unit 0 family iso address 49.0001.0003.0303.0300
set interfaces lo0 unit 0 family inet6 address 2001:db8:3:255::3/128
set policy-options policy-statement pplb then load-balance per-packet

```

```

set routing-options source-packet-routing srv6 locator myloc 2001:db8:0:a3::/64
set routing-options forwarding-table export pplb
set routing-options router-id 192.168.255.3
set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a3:1a32:: flavor psp
set protocols isis interface xe-0/0/0:0.0 node-link-protection
set protocols isis interface xe-0/0/0:0.0 point-to-point
set protocols isis interface xe-0/0/0:1.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a3:1a35:: flavor psp
set protocols isis interface xe-0/0/0:1.0 node-link-protection
set protocols isis interface xe-0/0/0:1.0 point-to-point
set protocols isis interface xe-0/0/0:2.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a3:1a36:: flavor psp
set protocols isis interface xe-0/0/0:2.0 node-link-protection
set protocols isis interface xe-0/0/0:2.0 point-to-point
set protocols isis interface xe-0/0/0:3.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a3:1a37:: flavor psp
set protocols isis interface xe-0/0/0:3.0 node-link-protection
set protocols isis interface xe-0/0/0:3.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srv6 locator myloc end-sid 2001:db8:0:a3:d31:: flavor
usd
set protocols isis level 1 disable

```

#### Router R4

```

set interfaces xe-0/0/0:0 description To_R0
set interfaces xe-0/0/0:0 mtu 4000
set interfaces xe-0/0/0:0 unit 0 family iso
set interfaces xe-0/0/0:0 unit 0 family inet6 address 2001:db8:4::2/64
set interfaces xe-0/0/0:2 description To_R1
set interfaces xe-0/0/0:2 mtu 4000
set interfaces xe-0/0/0:2 unit 0 family iso
set interfaces xe-0/0/0:2 unit 0 family inet6 address 2001:db8:14::2/64
set interfaces xe-0/0/0:3 description To_R2
set interfaces xe-0/0/0:3 mtu 4000
set interfaces xe-0/0/0:3 unit 0 family iso
set interfaces xe-0/0/0:3 unit 0 family inet6 address 2001:db8:24::2/64
set interfaces xe-0/0/1:0 description To_R5
set interfaces xe-0/0/1:0 mtu 4000
set interfaces xe-0/0/1:0 unit 0 family iso
set interfaces xe-0/0/1:0 unit 0 family inet6 address 2001:db8:25::1/64

```

```

set interfaces lo0 unit 0 family iso address 49.0001.0004.0404.0400
set interfaces lo0 unit 0 family inet6 address 2001:db8:4:255::4/128
set policy-options policy-statement pplb then load-balance per-packet
set routing-options source-packet-routing srv6 locator myloc 2001:db8:0:a4::/64
set routing-options forwarding-table export pplb
set routing-options router-id 192.168.255.4
set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a4:1a40:: flavor psp
set protocols isis interface xe-0/0/0:0.0 node-link-protection
set protocols isis interface xe-0/0/0:0.0 point-to-point
set protocols isis interface xe-0/0/0:2.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a4:1a41:: flavor psp
set protocols isis interface xe-0/0/0:2.0 node-link-protection
set protocols isis interface xe-0/0/0:2.0 point-to-point
set protocols isis interface xe-0/0/0:3.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a4:1a42:: flavor psp
set protocols isis interface xe-0/0/0:3.0 node-link-protection
set protocols isis interface xe-0/0/0:3.0 point-to-point
set protocols isis interface xe-0/0/1:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a4:1a45:: flavor psp
set protocols isis interface xe-0/0/1:0.0 node-link-protection
set protocols isis interface xe-0/0/1:0.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srv6 locator myloc end-sid 2001:db8:0:a4:d41:: flavor
usd
set protocols isis level 1 disable

```

## Router R5

```

set interfaces xe-0/0/0:0 description To_R1
set interfaces xe-0/0/0:0 mtu 4000
set interfaces xe-0/0/0:0 unit 0 family iso
set interfaces xe-0/0/0:0 unit 0 family inet6 address 2001:db8:15::2/64
set interfaces xe-0/0/0:1 description To_R2
set interfaces xe-0/0/0:1 mtu 4000
set interfaces xe-0/0/0:1 unit 0 family iso
set interfaces xe-0/0/0:1 unit 0 family inet6 address 2001:db8:25::2/64
set interfaces xe-0/0/0:2 description To_R3
set interfaces xe-0/0/0:2 mtu 4000
set interfaces xe-0/0/0:2 unit 0 family iso
set interfaces xe-0/0/0:2 unit 0 family inet6 address 2001:db8:35::2/64
set interfaces xe-0/0/0:3 description To_R4

```



```

set interfaces xe-0/0/0:3 mtu 4000
set interfaces xe-0/0/0:3 unit 0 family iso
set interfaces xe-0/0/0:3 unit 0 family inet6 address 2001:db8:45::2/64
set interfaces xe-0/0/1:0 description To_R6
set interfaces xe-0/0/1:0 mtu 4000
set interfaces xe-0/0/1:0 unit 0 family iso
set interfaces xe-0/0/1:0 unit 0 family inet6 address 2001:db8:56::1/64
set interfaces lo0 unit 0 family iso address 49.0001.0005.0505.0500
set interfaces lo0 unit 0 family inet6 address 2001:db8:5:255::5/128
set policy-options policy-statement pplb then load-balance per-packet
set routing-options source-packet-routing srv6 locator myloc 2001:db8:0:a5::/64
set routing-options forwarding-table export pplb
set routing-options router-id 192.168.255.5
set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a5:1a51:: flavor psp
set protocols isis interface xe-0/0/0:0.0 node-link-protection
set protocols isis interface xe-0/0/0:0.0 point-to-point
set protocols isis interface xe-0/0/0:1.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a5:1a52:: flavor psp
set protocols isis interface xe-0/0/0:1.0 node-link-protection
set protocols isis interface xe-0/0/0:1.0 point-to-point
set protocols isis interface xe-0/0/0:2.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a5:1a53:: flavor psp
set protocols isis interface xe-0/0/0:2.0 node-link-protection
set protocols isis interface xe-0/0/0:2.0 point-to-point
set protocols isis interface xe-0/0/0:3.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a5:1a54:: flavor psp
set protocols isis interface xe-0/0/0:3.0 node-link-protection
set protocols isis interface xe-0/0/0:3.0 point-to-point
set protocols isis interface xe-0/0/1:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a5:1a56:: flavor psp
set protocols isis interface xe-0/0/1:0.0 node-link-protection
set protocols isis interface xe-0/0/1:0.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srv6 locator myloc end-sid 2001:db8:0:a5:d51:: flavor
usd
set protocols isis level 1 disable

```

## Router R6

```

set interfaces xe-0/0/0:0 description To_R2
set interfaces xe-0/0/0:0 mtu 4000

```

```

set interfaces xe-0/0/0:0 unit 0 family iso
set interfaces xe-0/0/0:0 unit 0 family inet6 address 2001:db8:26::2/64
set interfaces xe-0/0/0:1 description To_R3
set interfaces xe-0/0/0:1 mtu 4000
set interfaces xe-0/0/0:1 unit 0 family iso
set interfaces xe-0/0/0:1 unit 0 family inet6 address 2001:db8:36::2/128
set interfaces xe-0/0/0:2 description To_R5
set interfaces xe-0/0/0:2 mtu 4000
set interfaces xe-0/0/0:2 unit 0 family iso
set interfaces xe-0/0/0:2 unit 0 family inet6 address 2001:db8:56::2/128
set interfaces xe-0/0/0:3 description To_R7
set interfaces xe-0/0/0:3 mtu 4000
set interfaces xe-0/0/0:3 unit 0 family iso
set interfaces xe-0/0/0:3 unit 0 family inet6 address 2001:db8:67::1/128
set interfaces lo0 unit 0 family iso address 49.0001.0006.0606.0600
set interfaces lo0 unit 0 family inet6 address 2001:db8:6:255::6/128
set policy-options policy-statement pplb then load-balance per-packet
set routing-options source-packet-routing srv6 locator myloc 2001:db8:0:a6::/64
set routing-options forwarding-table export pplb
set routing-options router-id 192.168.255.6
set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a6:1a62:: flavor psp
set protocols isis interface xe-0/0/0:0.0 node-link-protection
set protocols isis interface xe-0/0/0:0.0 point-to-point
set protocols isis interface xe-0/0/0:1.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a6:1a63:: flavor psp
set protocols isis interface xe-0/0/0:1.0 node-link-protection
set protocols isis interface xe-0/0/0:1.0 point-to-point
set protocols isis interface xe-0/0/0:2.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a6:1a65:: flavor psp
set protocols isis interface xe-0/0/0:2.0 node-link-protection
set protocols isis interface xe-0/0/0:2.0 point-to-point
set protocols isis interface xe-0/0/0:3.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a6:1a67:: flavor psp
set protocols isis interface xe-0/0/0:3.0 node-link-protection
set protocols isis interface xe-0/0/0:3.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srv6 locator myloc end-sid 2001:db8:0:a6:d61:: flavor
usd
set protocols isis level 1 disable

```

## Router R7

```

set interfaces xe-0/0/0:0 description To_R3
set interfaces xe-0/0/0:0 mtu 4000
set interfaces xe-0/0/0:0 unit 0 family iso
set interfaces xe-0/0/0:0 unit 0 family inet6 address 2001:db8:37::2/64
set interfaces xe-0/0/0:1 description To_R6
set interfaces xe-0/0/0:1 mtu 4000
set interfaces xe-0/0/0:1 unit 0 family iso
set interfaces xe-0/0/0:1 unit 0 family inet6 address 2001:db8:67::2/128
set interfaces xe-0/0/0:2 description To_CE2
set interfaces xe-0/0/0:2 unit 0 family inet address 172.16.20.2/24
set interfaces xe-0/0/0:2 unit 0 family iso
set interfaces lo0 unit 0 family iso address 49.0001.0007.0707.0700
set interfaces lo0 unit 0 family inet6 address 2001:db8:20:255::20/32
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement CE2_v4 term 1 from protocol direct
set policy-options policy-statement CE2_v4 term 1 from route-filter 172.16.20.0/24 exact
set policy-options policy-statement CE2_v4 term 1 then next-hop next-hop self
set policy-options policy-statement CE2_v4 term 1 then accept
set routing-options source-packet-routing srv6 locator myloc 2001:db8:0:a7::/64
set routing-options forwarding-table export pplb
set routing-options router-id 172.16.255.20
set routing-options autonomous-system 65550
set protocols bgp group to-R2RRv6 type internal
set protocols bgp group to-R2RRv6 local-address 2001:db8:20:255::20
set protocols bgp group to-R2RRv6 neighbor 2001:db8:2:255::2 family inet unicast extended-nexthop
set protocols bgp group to-R2RRv6 export CE2_v4
set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a7:1a73:: flavor psp
set protocols isis interface xe-0/0/0:0.0 node-link-protection
set protocols isis interface xe-0/0/0:0.0 point-to-point
set protocols isis interface xe-0/0/0:1.0 level 2 srv6-adjacency-segment protected locator myloc
end-x-sid 2001:db8:0:a7:1a76:: flavor psp
set protocols isis interface xe-0/0/0:1.0 node-link-protection
set protocols isis interface xe-0/0/0:1.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srv6 locator myloc end-sid 2001:db8:0:a7:d71:: flavor
usd
set protocols isis level 1 disable

```

## Configuring Router R0

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see [Using the CLI Editor in Configuration Mode](#) in the [CLI User Guide](#).

To configure SRv6 network programming to support IPv4 tunnels over a IPv6 core, perform the following steps on the R0 router:

### Step-by-Step Procedure

1. Configure the device interfaces to enable IP transport.

```
[edit]
user@R0# set interfaces xe-0/0/0:0 description To_R1_1
user@R0# set interfaces xe-0/0/0:0 vlan-tagging
user@R0# set interfaces xe-0/0/0:0 unit 0 vlan-id 1
user@R0# set interfaces xe-0/0/0:0 unit 0 family inet address 10.11.1.1/24
user@R0# set interfaces xe-0/0/0:0 unit 0 family iso
user@R0# set interfaces xe-0/0/0:0 unit 0 family inet6 address 2001:db8:1001::1/32
user@R0# set interfaces xe-0/0/0:2 description To_R4_1
user@R0# set interfaces xe-0/0/0:2 vlan-tagging
user@R0# set interfaces xe-0/0/0:2 unit 0 vlan-id 1
user@R0# set interfaces xe-0/0/0:2 unit 0 family inet address 10.21.1.1/24
user@R0# set interfaces xe-0/0/0:2 unit 0 family iso
user@R0# set interfaces xe-0/0/0:2 unit 0 family inet6 address 2001:db8:2021::1/32
user@R0# set interfaces xe-0/0/1:0 description to_RT
user@R0# set interfaces xe-0/0/1:0 vlan-tagging
user@R0# set interfaces xe-0/0/1:0 unit 1 vlan-id 1
user@R0# set interfaces xe-0/0/1:0 unit 1 family inet address 172.20.1.1/24
user@R0# set interfaces xe-0/0/1:0 unit 1 family iso
user@R0# set interfaces xe-0/0/1:0 unit 1 family inet6 address 2001:db8::20:1:1:1/120
user@R0# set interfaces xe-0/0/1:0 unit 4 vlan-id 4
user@R0# set interfaces xe-0/0/1:0 unit 4 family inet address 172.20.2.1/24
user@R0# set interfaces xe-0/0/1:0 unit 4 family iso
user@R0# set interfaces xe-0/0/1:0 unit 4 family inet6 address 2001:db8::20:2:1:1/120
```

2. Configure the loopback interface with IPv4 and IPv6 addresses that is used as router ID for BGP sessions.

```
[edit]
user@R0# set interfaces lo0 unit 0 family inet address 192.168.0.10/32
user@R0# set interfaces lo0 unit 0 family iso address 49.0001.000a.0a0a.0a00
user@R0# set interfaces lo0 unit 0 family inet6 address 2001:db8::10:10:10:10/32
```

3. Configure the router ID and autonomous system (AS) number to propagate routing information within a set of routing devices that belong to the same AS.

```
[edit]
user@R0# set routing-options router-id 10.10.10.10
user@R0# set routing-options autonomous-system 65550
```

4. Enable SRv6 globally and the locator address to indicate the SRv6 capability of the router. SRv6 SID is an IPv6 address that consists of the locator and a function. The routing protocols advertise the locator addresses.

```
[edit]
user@R0# set routing-options source-packet-routing srv6 locator myloc 2001:db8:0:a0::/64
```

5. Configure the End-SID function for the prefix segments. Specify a flavor, that is the behavior of the End-SID function as per your network requirements. Penultimate Segment Pop (PSP), Ultimate Segment Pop (USP), and Ultimate Segment Decapsulation (USD) are the three available flavors for SRv6 functions.



**NOTE:** Ensure that the locator and the End-SID are in the same subnet to avoid a commit error.

```
[edit]
user@R0# set protocols isis source-packet-routing srv6 locator myloc end-sid
2001:db8:0:a0:d01:: flavor usd
user@R0# set protocols isis source-packet-routing srv6 locator myloc1 end-sid
2001:db8:0:a10:d01:: flavor usd
user@R0# set protocols isis source-packet-routing srv6 locator myloc2 end-sid
2001:db8:0:a20:d01:: flavor usd
user@R0# set protocols isis source-packet-routing srv6 locator myloc3 end-sid
```

```

2001:db8:0:a30:d01:: flavor usd
user@R0# set protocols isis source-packet-routing srv6 locator myloc4 end-sid
2001:db8:0:a40:d01:: flavor usp
user@R0# set protocols isis source-packet-routing srv6 locator myloc4 end-sid
2001:db8:0:a40:d01:: flavor usd
user@R0# set protocols isis level 1 disable

```

6. Configure End-X-SID function on the point-to-point (P2P) interface for the adjacency segments. Specify one or more flavor for the End-X-SID.



**NOTE:** Ensure that the Locator and End-X-SID are in the same subnet to avoid a commit error. You must enable SRv6 and configure the locator at the [edit routing-options] before mapping locators to interfaces. Whenever you configure an srv6-adjacency-segment you must also configure the related locator under the protocols isis source-packet-routing srv6 locator hierarchy, as shown in step 5.

```

[edit]
user@R0# set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected
locator myloc end-x-sid 2001:db8:0:a0:1a01:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected
locator myloc1 end-x-sid 2001:db8:0:a10:1a01:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected
locator myloc2 end-x-sid 2001:db8:0:a20:1a01:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected
locator myloc3 end-x-sid 2001:db8:0:a30:1a01:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:0.0 level 2 srv6-adjacency-segment protected
locator myloc4 end-x-sid 2001:db8:0:a40:1a01:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:0.0 node-link-protection
user@R0# set protocols isis interface xe-0/0/0:0.0 point-to-point

```

7. Configure SRv6 options for the adjacency segment of the LAN interface xe-0/0/0:2.0. Specify a flavor as per your network requirements. Penultimate Segment Pop (PSP), Ultimate Segment Pop (USP), and Ultimate Segment Decapsulation (USP) are the three available flavors for the SRv6 adjacency segment.



**NOTE:** Ensure that the Locator and End-X-Sid are in the same subnet to avoid a commit error. You must enable SRv6 and configure the locator at the [edit routing-options] before mapping locators to interfaces.

```
[edit]
user@R0# set protocols isis interface xe-0/0/0:2.0 level 2 lan-neighbor 0100.0404.0404 srv6-
adjacency-segment unprotected locator myloc end-x-sid 2001:db8:0:a0:1a04:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:2.0 level 2 lan-neighbor 0100.0404.0404 srv6-
adjacency-segment unprotected locator myloc1 end-x-sid 2001:db8:0:a10:1a04:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:2.0 level 2 lan-neighbor 0100.0404.0404 srv6-
adjacency-segment unprotected locator myloc2 end-x-sid 2001:db8:0:a20:1a04:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:2.0 level 2 lan-neighbor 0100.0404.0404 srv6-
adjacency-segment unprotected locator myloc3 end-x-sid 2001:db8:0:a30:1a04:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:2.0 level 2 lan-neighbor 0100.0404.0404 srv6-
adjacency-segment unprotected locator myloc4 end-x-sid 2001:db8:0:a40:1a04:: flavor usd
user@R0# set protocols isis interface xe-0/0/0:2.0 node-link-protection
user@R0# set protocols isis interface xe-0/0/1:0.1
user@R0# set protocols isis interface fxp0.0 disable
user@R0# set protocols isis interface lo0.0 passive
```

8. Configure BGP on the core-facing interface to establish internal peering sessions.

```
[edit]
user@R0# set protocols bgp group to-PEv6 type internal
user@R0# set protocols bgp group to-PEv6 local-address abcd::10:10:10:10
user@R0# set protocols bgp group to-PEv6 neighbor abcd::2:2:2:2 family inet unicast
extended-nexthop
user@R0# set protocols bgp group to-PE2 type internal
user@R0# set protocols bgp group to-PE2 local-address 10.10.10.10
user@R0# set protocols bgp group to-PE2 neighbor 2.2.2.2 family inet6 unicast
user@R0# set protocols bgp group to-PE2 neighbor 2.2.2.2 family inet6-vpn unicast
```

9. Define a policy to load balance packets.

```
[edit]
user@R0# set policy-options policy-statement pplb then load-balance per-packet
```

10. Apply the per-packet policy to enable load balancing of traffic.

```
[edit]
user@R0# set routing-options forwarding-table export pplb
```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, `show policy-options`, and `show routing-options` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@R0# show interfaces
xe-0/0/0:0 {
  description To_R1;
  mtu 4000;
  unit 0 {
    family iso;
    family inet6 {
      address 2001:db8:1::1/64;
    }
  }
}
xe-0/0/0:2 {
  description To_R4;
  mtu 4000;
  unit 0 {
    family iso;
    family inet6 {
      address 2001:db8:4::1/64;
    }
  }
}
xe-0/0/1:0 {
  description To_CE1;
  unit 0 {
    family inet {
      address 172.16.10.2/24;
    }
    family iso;
  }
}
```



```

}
lo0 {
    unit 0 {
        family iso {
            address 49.0001.000a.0a0a.0a00;
        }
        family inet6 {
            address 2001:db8:10:255::10/128;
        }
    }
}
}

```

```

[edit]
user@R0# show protocols
bgp {
    group to-R2RRv6 {
        type internal;
        local-address 2001:db8:10:255::10;
        export CE1_v4;
        neighbor 2001:db8:2:255::2 {
            family inet {
                unicast {
                    extended-nexthop;
                }
            }
        }
    }
}
isis {
    interface xe-0/0/0:0.0 {
        level 2 {
            srv6-adjacency-segment {
                protected {
                    locator myloc {
                        end-x-sid 2001:db8:0:a0:1a01:: {
                            flavor psp;
                        }
                    }
                }
            }
        }
    }
}

```

```

        node-link-protection;
        point-to-point;
    }
    interface xe-0/0/0:2.0 {
        level 2 {
            srv6-adjacency-segment {
                protected {
                    locator myloc {
                        end-x-sid 2001:db8:0:a0:1a04:: {
                            flavor psp;
                        }
                    }
                }
            }
        }
        node-link-protection;
        point-to-point;
    }
    interface lo0.0 {
        passive;
    }
    source-packet-routing {
        srv6 {
            locator myloc {
                end-sid 2001:db8:0:a0:d01:: {
                    flavor {
                        usd;
                    }
                }
            }
        }
    }
    level 1 disable;
}

```

```

[edit]
user@R0# show policy-options
policy-statement CE1_v4 {
    term 1 {
        from {
            protocol direct;

```

```

        route-filter 172.16.10.0/24 exact;
    }
    then {
        next-hop 2001:db8:0:a0:d01::;
        accept;
    }
}
}
policy-statement pplb {
    then {
        load-balance per-packet;
    }
}
}

```

```

[edit]
user@R0# show routing-options
source-packet-routing {
    srv6 {
        locator myloc 2001:db8:0:a0::/64;
    }
}
forwarding-table {
    export pplb;
}
router-id 172.16.255.10;
autonomous-system 65550;

```

When done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Verifying IS-IS Adjacency and IBGP Session | 590](#)
- [Verify SRv6 is Enabled | 591](#)
- [Verify the SRv6 End-X-SID Configuration | 592](#)
- [Verifying the Locator Route is Installed | 593](#)
- [Verifying the End-X-SID Route is Installed | 594](#)

- [Verifying the End-SID Route is Installed | 595](#)
- [Verify the SRv6 Configuration in the IS-IS Database | 598](#)
- [Verifying the Route to CE2 Uses an SRv6 Tunnel | 600](#)
- [Test IPv4 Connectivity Between CE1 and CE2 | 600](#)

Confirm that the configuration is working properly.

## Verifying IS-IS Adjacency and IBGP Session

### Purpose

Verify IS-IS adjacencies and IBGP session at R2. R2 is chosen for this task because it has 5 adjacencies and also serves as the router reflector for the BGP control plane.



**NOTE:** Its a good idea to confirm the IS-IS adjacencies on all routers before proceeding to the remaining verification steps. A successful SRv6 deployment requires that the interior gateway protocol is operational on all nodes.

### Action

From operational mode, run the **show isis adjacency** command on router R2.

```
user@R2> show isis adjacency
```

Interface	System	L State	Hold (secs)	SNPA
xe-0/0/0:0.0	R1	2 Up	26	
xe-0/0/0:1.0	R3	2 Up	25	
xe-0/0/0:2.0	R4	2 Up	25	
xe-0/0/0:3.0	R5	2 Up	24	
xe-0/0/1:0.0	R6	2 Up	18	

From operational mode, run the **show bgp summary** command on router R2.

```
user@R2> show bgp summary
Threading mode: BGP I/O
Default eBGP mode: advertise - accept, receive - accept
Groups: 1 Peers: 2 Down peers: 0
```

Table	Tot Paths	Act Paths	Suppressed	History	Damp	State	Pending
inet.0	2	2	0	0	0	0	0
inet6.0	0	0	0	0	0	0	0
Peer	AS	InPkt	OutPkt	OutQ	Flaps	Last Up/Dwn	State #Active/
Received/Accepted/Damped...							
2001:db8:10:255::10	65550	3101	3092	0	0	23:14:18	Establ
inet.0: 1/1/1/0							
2001:db8:20:255::20	65550	3091	3080	0	0	23:10:10	Establ
inet.0: 1/1/1/0							

## Meaning

The output confirms the expected IS-IS adjacency count for the R2 router. It also confirms that R2 has established IPv6 based BGP sessions to both the R0 and R7 routers.

## Verify SRv6 is Enabled

### Purpose

Verify that SRv6 is enabled with a locator, End-SID, and flavor on Router R0.

### Action

From operational mode, run the **show isis overview** command on Router R0.

```

user@R0> show isis overview
Instance: master
  Router ID: 172.16.255.10
  IPv6 Router ID: 2001:db8:1::1
  Hostname: R0
  Sysid: 0100.0a0a.0a0a
  Areaid: 49.00
  Adjacency holddown: enabled
  Maximum Areas: 3
  LSP life time: 1200
  Attached bit evaluation: enabled
  SPF delay: 200 msec, SPF holddown: 5000 msec, SPF rapid runs: 3
  IPv4 is enabled, IPv6 is enabled
  Traffic engineering: enabled

```

```

Restart: Disabled
  Helper mode: Enabled
Layer2-map: Disabled
Source Packet Routing (SPRING): Enabled
  Node Segments: Disabled
SRv6: Enabled
  Locator: 2001:db8:0:a0::/64, Algorithm: 0
  END-SID: 2001:db8:0:a0:d01::, Flavor: USD
Post Convergence Backup: Disabled
Level 1
  Internal route preference: 15
  External route preference: 160
  Prefix export count: 0
  Wide metrics are enabled, Narrow metrics are enabled
  Source Packet Routing is enabled
Level 2
  Internal route preference: 18
  External route preference: 165
  Prefix export count: 0
  Wide metrics are enabled, Narrow metrics are enabled
  Source Packet Routing is enabled

```

## Meaning

The configured SRv6 locator SRv6: Enabled Locator: 2001:db8:0:a0::/64, Algorithm: 0 and , End-SID and flavor END-SID: 2001:db8:0:a0:d01::, Flavor: USD are displayed in the output.

## Verify the SRv6 End-X-SID Configuration

### Purpose

Verify that an End-X-SID function and flavor are configured on R0.

### Action

From operational mode, run the **show isis adjacency detail** command on Router R0.

```

user@R0> show isis adjacency detail
R1
  Interface: xe-0/0/0:0.0, Level: 2, State: Up, Expires in 19 secs
  Priority: 0, Up/Down transitions: 1, Last transition: 03:51:48 ago

```

```

Circuit type: 2, Speaks: IP, IPv6
Topologies: Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 192.168.255.1
IPv6 addresses: fe80::2e6b:f5ff:fedb:e800
IPv6 Global Interface Address: 2001:db8:1::2
Level 2 SRv6 protected END-X-SID: 2001:db8:0:a0:1a01::
  Flavor: PSP, Flags: B-P, Algorithm: 0

```

R4

```

Interface: xe-0/0/0:2.0, Level: 2, State: Up, Expires in 20 secs
Priority: 0, Up/Down transitions: 1, Last transition: 03:48:04 ago
Circuit type: 2, Speaks: IP, IPv6
Topologies: Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 192.168.255.4
IPv6 addresses: fe80::2e6b:f5ff:feb4:4000
IPv6 Global Interface Address: 2001:db8:4::2
Level 2 SRv6 protected END-X-SID: 2001:db8:0:a0:1a04::
  Flavor: PSP, Flags: B-P, Algorithm: 0

```

## Meaning

The field SRv6 protected END-X-SID: 2001:db8:0:a0:1a01:: indicates that End-X-SID function with Flavor PSP has been configured on router R0 for the interface used to attach to R1. Similar output is confirmed for the interface connected to R4, which uses a different End-X-SID.

## Verifying the Locator Route is Installed

### Purpose

Verify that the locator route has been installed.

### Action

From operational mode, run the **show route 2001:db8:0:a0::/64 detail** command on router R0.

```

user@R0> show route 2001:db8:0:a0::/64 detail
inet6.0: 75 destinations, 75 routes (75 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

```

```

2001:db8:0:a0::/64*[IS-IS/18] 3d 19:03:16, metric 0
      Reject

user@R0> show route 2001:db8:0:a0::/64 detail
inet6.0: 45 destinations, 45 routes (45 active, 0 holddown, 0 hidden)
2001:db8:0:a0::/64 (1 entry, 1 announced)
  *IS-IS Preference: 18
    Level: 2
    Next hop type: Reject, Next hop index: 0
    Address: 0xc54526c
    Next-hop reference count: 2
    State: <Active Int OpaqueData>
    Local AS: 65550
    Age: 22:15:32 Metric: 0
    Validation State: unverified
    ORR Generation-ID: 0
    Task: IS-IS
    Announcement bits (2): 0-KRT 5-Resolve tree 5
    AS path: I
  . . .

```

## Meaning

The output confirms the locator route `2001:db8:0:a0::/64*[IS-IS/18]` is installed in the `inet6.0` table.

## Verifying the End-X-SID Route is Installed

### Purpose

To display the configured End-X-SID route information that is applied at the interface.

### Action

From operational mode, run the **show route 2001:db8:0:a0:1a01::** command on Router R0.

```

user@R0> show route 2001:db8:0:a0:1a01::
inet6.0: 45 destinations, 45 routes (45 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8:0:a0::1a01/128

```



```
*[IS-IS/18] 04:33:42, metric 0
> to fe80::2e6b:f5ff:fedb:e800 via xe-0/0/0:0.0
```

## Meaning

The output confirms the End-X-SID route 2001:db8:0:a0::1a01/128 is installed in the inet.6.0 routing table.

## Verifying the End-SID Route is Installed

### Purpose

Verify that the End-SID routes for all routers in the SRv6 domain are installed in the inet6.3 table at Router R0.

### Action

From operational mode, run the **show route table inet6.3 protocol isis** command on Router R0 to see all End-SIDs the router has learned. Then display detailed information about the End-SID associated with the R7 router.

```
user@R0> show route table inet6.3 protocol isis
inet6.3: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8:0:a1::d11/128
    *[SRV6-ISIS/14] 04:39:22, metric 10
    > to fe80::2e6b:f5ff:fedb:e800 via xe-0/0/0:0.0, SRV6-Tunnel, Dest:
2001:db8:0:a1:d11::
2001:db8:0:a2::d21/128
    *[SRV6-ISIS/14] 04:35:38, metric 20
    to fe80::2e6b:f5ff:fedb:e800 via xe-0/0/0:0.0, SRV6-Tunnel, Dest:
2001:db8:0:a2:d21::
    > to fe80::2e6b:f5ff:feb4:4000 via xe-0/0/0:2.0, SRV6-Tunnel, Dest:
2001:db8:0:a2:d21::
2001:db8:0:a3::d31/128
    *[SRV6-ISIS/14] 04:35:38, metric 30
    to fe80::2e6b:f5ff:fedb:e800 via xe-0/0/0:0.0, SRV6-Tunnel, Dest:
2001:db8:0:a3:d31::
    > to fe80::2e6b:f5ff:feb4:4000 via xe-0/0/0:2.0, SRV6-Tunnel, Dest:
2001:db8:0:a3:d31::
2001:db8:0:a4::d41/128
```

```

                *[SRV6-ISIS/14] 04:35:38, metric 10
                > to fe80::2e6b:f5ff:feb4:4000 via xe-0/0/0:2.0, SRV6-Tunnel, Dest:
2001:db8:0:a4:d41::
2001:db8:0:a5::d51/128
                *[SRV6-ISIS/14] 04:35:01, metric 20
                to fe80::2e6b:f5ff:fedb:e800 via xe-0/0/0:0.0, SRV6-Tunnel, Dest:
2001:db8:0:a5:d51::
                > to fe80::2e6b:f5ff:feb4:4000 via xe-0/0/0:2.0, SRV6-Tunnel, Dest:
2001:db8:0:a5:d51::
2001:db8:0:a6::d61/128
                *[SRV6-ISIS/14] 04:34:32, metric 30
                to fe80::2e6b:f5ff:fedb:e800 via xe-0/0/0:0.0, SRV6-Tunnel, Dest:
2001:db8:0:a6:d61::
                > to fe80::2e6b:f5ff:feb4:4000 via xe-0/0/0:2.0, SRV6-Tunnel, Dest:
2001:db8:0:a6:d61::
2001:db8:0:a7::d71/128
                *[SRV6-ISIS/14] 04:33:00, metric 40
                to fe80::2e6b:f5ff:fedb:e800 via xe-0/0/0:0.0, SRV6-Tunnel, Dest:
2001:db8:0:a7:d71::
                > to fe80::2e6b:f5ff:feb4:4000 via xe-0/0/0:2.0, SRV6-Tunnel, Dest:
2001:db8:0:a7:d71::

```

```

user@R0> show route 2001:db8:0:a7::d71/128 detail

```

```

inet6.3: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)

```

```

2001:db8:0:a7::d71/128 (1 entry, 1 announced)

```

```

    *SRV6-ISIS Preference: 14

```

```

        Level: 2

```

```

        Next hop type: List, Next hop index: 1048577

```

```

        Address: 0xdb8deb4

```

```

        Next-hop reference count: 6

```

```

        Next hop: ELNH Address 0xc5462d4 weight 0x1

```

```

            Next hop type: Chain, Next hop index: 582

```

```

            Address: 0xc5462d4

```

```

            Next-hop reference count: 1

```

```

            Next hop: ELNH Address 0xc545bcc

```

```

SRV6-Tunnel: Reduced-SRH Encap-mode

```

```

Src: 2001:db8:1::1 Dest: 2001:db8:0:a7:d71::

```

```

Segment-list[0] 2001:db8:0:a7:d71::

```

```

            Next hop type: Router, Next hop index: 580

```

```

            Address: 0xc545bcc

```

```

            Next-hop reference count: 9

```

```

            Next hop: fe80::2e6b:f5ff:fedb:e800 via xe-0/0/0:0.0 weight 0x1

```

```

Next hop: ELNH Address 0xc546338 weight 0x1, selected
  Next hop type: Chain, Next hop index: 583
  Address: 0xc546338
  Next-hop reference count: 1
  Next hop: ELNH Address 0xc545f50
  SRV6-Tunnel: Reduced-SRH Encap-mode
  Src: 2001:db8:1::1 Dest: 2001:db8:0:a7:d71::
  Segment-list[0] 2001:db8:0:a7:d71::
    Next hop type: Router, Next hop index: 581
    Address: 0xc545f50
    Next-hop reference count: 9
    Next hop: fe80::2e6b:f5ff:feb4:4000 via xe-0/0/0:2.0 weight 0x1
State: <Active NoReadvrt Int OpaqueData>
Local AS: 65550
Age: 4:35:43    Metric: 40
Validation State: unverified
ORR Generation-ID: 0
Task: IS-IS
Announcement bits (3): 0-Resolve tree 2 1-Resolve tree 5 2-Resolve_IGP_FRR task
AS path: I
Session-IDs associated:
Session-id: 322 Version: 0

```

## Meaning

The output confirms that Router R0 has learned End-SIDs, that is, 2001:db8:0:a1::d11/128 and 2001:db8:0:a2::d21/128, from all other routers in the topology. Note the End-SIDs have been installed in the inet6.3 table. The detailed output for the End-SID advertised by R7 2001:db8:0:a7:d71:: confirms an SRv6 tunnel has been established between Router R0 and Router R7.

Note that the segment list is populated with the End-SID value configured on the Router R7. Recall that all End-SIDs in this example are configured with the Ultimate Segment Decapsulate (USD) flavor. It's the combination of a local End-SID and the associated USD flavor that tells R7 it's the egress of the IPv6 tunnel. Upon receipt R7 decapsulates the IPv4 packet and routes it according to the IPv4 destination address.

## Verify the SRv6 Configuration in the IS-IS Database

### Purpose

Display the IS-IS database to verify the End-SID and flavor configured at Router R7. In this example the command is executed on Router R0. Similar output is expected on all router because the IS-IS database is replicated to all nodes.

### Action

From operational mode, run the **show isis database R7.00-00 extensive** command on Router R0.

```
user@R0> show isis database R.00-00 extensive
IS-IS level 1 link-state database:

IS-IS level 2 link-state database:

R7.00-00 Sequence: 0x31f, Checksum: 0x2ce6, Lifetime: 904 secs
  IS neighbor: R3.00                                Metric:      10
    Two-way fragment: R3.00-00, Two-way first fragment: R3.00-00
  IS neighbor: R6.00                                Metric:      10
    Two-way fragment: R6.00-00, Two-way first fragment: R6.00-00
  V6 prefix: 2001:db8::/32                          Metric:      0 Internal Up
  V6 prefix: 2001:db8:0:a7::/64                     Metric:      0 Internal Up
  V6 prefix: 2001:db8:20:255::20/128                 Metric:      0 Internal Up
  V6 prefix: 2001:db8:37::/64                       Metric:     10 Internal Up
  V6 prefix: 2001:db8:67::2/128                     Metric:     10 Internal Up

Header: LSP ID: R7.00-00, Length: 445 bytes
  Allocated length: 746 bytes, Router ID: 172.16.255.20
  Remaining lifetime: 904 secs, Level: 2, Interface: 360
  Estimated free bytes: 0, Actual free bytes: 301
  Aging timer expires in: 904 secs
  Protocols: IP, IPv6

Packet: LSP ID: R7.00-00, Length: 445 bytes, Lifetime : 1192 secs
  Checksum: 0x2ce6, Sequence: 0x31f, Attributes: 0x3 <L1 L2>
  NLPID: 0x83, Fixed length: 27 bytes, Version: 1, Sysid length: 0 bytes
  Packet type: 20, Packet version: 1, Max area: 0

TLVs:
  Area address: 49.00 (2)
```

```

LSP Buffer Size: 1492
Speaks: IP
Speaks: IPV6
IP router id: 172.16.255.20
IP address: 172.16.255.20
IPv6 TE Router ID: 2001:db8:20:255::20
Hostname: R7
SRv6 Locator: 2001:db8:0:a7::/64, Metric: 0, MTID: 0, Flags: 0x0, Algorithm: 0
  SRv6 SID: 2001:db8:0:a7:d71::, Flavor: USD
IPv6 prefix: 2001:db8:20:255::20/128 Metric 0 Up
IPv6 prefix: 2001:db8::/32 Metric 0 Up
IPv6 prefix: 2001:db8:0:a7::/64 Metric 0 Up
IPv6 prefix: 2001:db8:37::/64 Metric 10 Up
IPv6 prefix: 2001:db8:67::2/128 Metric 10 Up
Router Capability: Router ID 172.16.255.20, Flags: 0x00
  SPRING Algorithm - Algo: 0
  SRv6 Capability - Flags: 0
  Node MSD Advertisement Sub-TLV:Type: 23, Length: 10
  SRv6 Maximum Segments Left MSD:Type: 41, Value: 6
  SRv6 Maximum Pop MSD:Type: 42, Value: 7
  SRv6 Maximum Insert MSD:Type: 43, Value: 5
  SRv6 Maximum Encap MSD:Type: 44, Value: 6
  SRv6 Maximum End D MSD:Type: 45, Value: 6
  IPv6 TE Router Id: 2001:db8:20:255::20
IS neighbor: R6.00, Internal, Metric: default 10
IS neighbor: R3.00, Internal, Metric: default 10
Extended IS Reachability TLV, Type: 22, Length: 174
IS extended neighbor: R6.00, Metric: default 10 SubTLV len: 76
  IPv6 address: 2001:db8:67::2
  Neighbor's IP address: 192.168.255.6
  Neighbor's IPv6 address: 2001:db8:67::1
  Local interface index: 361, Remote interface index: 364
P2P SRV6 END-X-SID:2001:db8:0:a7:1a76::, Flags:B-P, Weight:0, Algorithm:0
  Flags:0xa0(B:1,S:0,P:1), Flavor: PSP
IS extended neighbor: R3.00, Metric: default 10 SubTLV len: 76
  IPv6 address: 2001:db8:37::2
  Neighbor's IP address: 192.168.255.3
  Neighbor's IPv6 address: 2001:db8:37::1
  Local interface index: 360, Remote interface index: 336
P2P SRV6 END-X-SID:2001:db8:0:a7:1a73::, Flags:B-P, Weight:0, Algorithm:0
  Flags:0xa0(B:1,S:0,P:1), Flavor: PSP
No queued transmissions

```

## Meaning

The presence of SRv6 SID: 2001:db8:0:a7:d71:: with Flavor: USD confirms that SRv6 is enabled with a SID decapsulate flavor on the R7 router. The output also shows that the interfaces at R7 have been configured for TI-LFA protection using a PSP flavor.

## Verifying the Route to CE2 Uses an SRv6 Tunnel

### Purpose

Display the route to the IPv4 subnet at R7 to confirm the next hop points to an SRv6 tunnel.

### Action

From operational mode, run the **show route 172.16.20.0/24** command on router R0.

```
user@R0> show route 172.16.20.0/24

inet.0: 36 destinations, 36 routes (36 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.20.0/24    *[BGP/170] 05:20:58, localpref 100, from 2001:db8:2:255::2
                  AS path: I, validation-state: unverified
                  to fe80::2e6b:f5ff:fedb:e800 via xe-0/0/0:0.0, SRV6-Tunnel, Dest:
2001:db8:0:a7:d71::
                  > to fe80::2e6b:f5ff:feb4:4000 via xe-0/0/0:2.0, SRV6-Tunnel, Dest:
2001:db8:0:a7:d71::
```

## Meaning

The output confirms that R0 has learned the route to the 172.16.20.0/24 subnet through its BGP session to R2, which recall is configured as a route reflector in this example. The next hops confirm that an SRv6 tunnel to the R7 router has been installed for this route. Two next hops are available in keeping with their being two equal cost paths between the R0 and R7 routers in the example topology.

## Test IPv4 Connectivity Between CE1 and CE2

### Purpose

Generate pings to verify IPv4 connectivity between the CE devices over the IPv6 provider core.

## Action

From operational mode, run the **ping 172.16.20.2 source 172.16.10.2 count 2** command on router R0.

```
user@R0> ping 172.16.20.2 source 172.16.10.2 count 2
PING 172.16.20.2 (172.16.20.2): 56 data bytes
64 bytes from 172.16.20.2: icmp_seq=0 ttl=64 time=114.922 ms
64 bytes from 172.16.20.2: icmp_seq=1 ttl=64 time=89.558 ms

--- 172.16.20.2 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 89.558/102.240/114.922/12.682 ms
```

## Meaning

The output confirms IPv4 connectivity is working between the CE device networks. This provides verification that SRv6 tunneling over an IPv6 provider core is working properly in this example.

## RELATED DOCUMENTATION

*locator*

*srv6*

## Understand SRv6 Static Segment Identifier

### IN THIS SECTION

- [Benefits of Segment Routing for IPv6 Static Segment Identifier Configuration | 602](#)
- [Configuration Overview | 602](#)

The Segment Routing for IPv6 (SRv6) Static Segment Identifier (SID) Configuration feature enhances network traffic management by allowing static allocation of Segment Identifiers (SIDs) for micro and classic locators. This functionality is pivotal for environments where traditional Interior Gateway

Protocols (IGP) like IS-IS are not utilized, providing an alternative path computation method. By enabling the advertisement of these static SIDs over BGP, you can facilitate segment routing traffic engineering, improving both efficiency and flexibility. You gain the capability to configure micro node, adjacency, and classic END and END-X SIDs, with support for primary and backup interfaces and various flavors. Additionally, SRv6 Operations, Administration, and Maintenance (OAM) support, including ping and traceroute functionalities, bolsters operational and troubleshooting capabilities. With several CLI commands available for configuration, this feature ensures effective use of static SIDs while managing potential conflicts with dynamic routing protocols.

## Benefits of Segment Routing for IPv6 Static Segment Identifier Configuration

- Enhance network traffic management by allowing the use of static SIDs in environments lacking traditional IGP routing protocols, ensuring effective path computation and traffic engineering.
- Facilitate segment routing traffic engineering by enabling static SID advertisement over BGP, which improves network efficiency and flexibility.
- Provide robust operational support through SRv6 OAM features like ping and traceroute, aiding in effective network troubleshooting and maintenance.
- Support configuration for both primary and backup interfaces, ensuring network resilience and continuity in case of interface failures.
- Allow integration of static SIDs with dynamic tunnels for Layer 3 VPN services, supporting service resolution and maintaining network service quality.

## Configuration Overview

The SRv6 Static SID Configuration feature allows you to define static SIDs in your network, providing precise control over traffic management without the need for traditional IGP protocols like IS-IS. By configuring these static SIDs, you can advertise them over BGP, facilitating segment routing traffic engineering that optimizes path computation and network efficiency. The feature supports various SID types, including micro node and adjacency SIDs as well as classic END and END-X SIDs, offering flexibility in managing both primary and backup interfaces.

SRv6 manager supports micro-sid locators under the `[edit routing-options source-packet-routing srv6]` hierarchy. This supports allocation of static and dynamic node SIDs and adjacency SIDs with SRv6 by applications. The flavors configured for the locator is inherited by the micro SIDs configured under it. This configuration supports flavors such as penultimate segment pop (PSP), ultimate segment



decapsulation (USD), and ultimate segment pop (USP), which are essential for tailoring traffic flows according to specific network requirements. The following is the configuration hierarchy:

```
[edit routing-options]
user@host# set source-packet-routing srv6 locator locator-name micro-sid flavor ?
Possible completions:
+ apply-groups          Groups from which to inherit configuration data
+ apply-groups-except   Don't inherit configuration data from these groups
  none                  No flavor
  psp                   Penultimate segment pop of the SRH
  usd                   Ultimate segment decapsulation
  usp                   Ultimate segment pop of the SRH
```

You can allocate node SIDs for micro-SID locators using the following configuration. If IS-IS is configured, it installs remote micro node SIDs in the inet6.3 table. Remote micro node SIDs allocated by SRv6 manager are advertised through the BGP export policy and is installed in inet6.0 table.

```
[edit routing-options]
user@host# set source-packet-routing srv6 locator locator-name micro-sid node-sid
```

You can allocate adjacency SIDs for micro-SID locators. Static SRv6 routes for adjacency SIDs is updated on any interface events. When a primary or backup interface goes down, the corresponding next hop is removed from the route and is added back when the interface becomes available. For a static micro adjacency SID, you can configure up to eight interfaces as backup. Here is the configuration to allocate adjacency SIDs for micro-SID locators:

```
[edit routing-options]
user@host# set source-packet-routing srv6 locator locator-name micro-sid adjacency-sid
```

The following is a sample configuration of adjacency SIDs for micro-SID:

```
[edit routing-options]
user@host# set source-packet-routing srv6 locator loc1 2001:db8::/48 micro-sid adjacency-sid
0xf850 interface ge-0/0/0.0
Possible completions:
<[Enter]>          Execute this command
next-hop          Next hop for adjacency sid
backup           Marks the interface as protection for adjacency SID
|                Pipe through a command
```

The backup statement is used to indicate that the interface is going to act as a protection for the adjacency SID. This allows to form a non-weighted ECMP for the adjacency SID and in the event of primary interface going down, the PFE switches the traffic to backup.

The next-hop statement is required in the case of broadcast links where the next hop address must be explicitly mentioned for proper forwarding.

SRv6 manager supports allocation of static and dynamic END SIDs and END-X SIDs for classic locators. You can now allocate static END SIDs and END-X SIDs with SRv6 as the application. Flavors for classic SIDs can be configured as follows:

```
[edit routing-options]
user@host# set source-packet-routing srv6 locator loc1 2001:db8::/48 end-sid 5000::1
Possible completions:
  <[Enter]>                Execute this command
  > flavor                  Configure end-SID flavor
  |                        Pipe through a command
```

You can allocate END SIDs for classic locators along with flavors PSP, USD and USP. If IS-IS is configured, it installs remote END SIDs in the inet6.3 table. Remote END SIDs allocated by SRv6 manager is advertised through BGP export policy and is installed in inet6.0 table.

You can allocate END-X SIDs for classic locators along with flavors PSP, USD and USP. Static SRv6 routes for END-X SIDs is updated on any interface events. Like micro-adjacency SIDs, when the primary or backup interface goes down, the corresponding next hop is removed from the route and is added back once the interface becomes available. Like micro adjacency SIDs, END-X SIDs supports backup interfaces and next-hop. For a static END-X SID, up to eight interfaces can be configured as backup.

The following is a sample configuration of END-X SIDs:

```
[edit routing-options]
user@host# set source-packet-routing srv6 locator loc1 2001:db8::/48 end-x-sid 0xf850 interface
ge-0/0/0.0
Possible completions:
  <[Enter]>                Execute this command
  next-hop                Next hop for END-X SID
  backup                  Marks the interface as protection for END-X SID
  |                      Pipe through a command
```

It is possible for both IS-IS and SRv6 manager to configure static SIDs for the same locator. You need to consider the following while configuring both IS-IS and static SIDs:

- When both IS-IS and SRv6 manager configures the same static micro adjacency SID or classic adjacency SID or classic node SID, a commit check fails due to duplicate allocation.
- Micro node SID can be configured by both IS-IS and SRv6 manager without any commit check failures. In this case, the IS-IS' SID route will have more preference over the SRv6 route.
- It is possible for IS-IS to configure non-zero algo SIDs but the SRv6 manager only installs zero algo SIDs in the inet6.0 table even though algorithm is configured for the locator. IS-IS installs the non-zero algo SIDs in the inet6color.0 table while using the same locator as SRv6 manager.

Moreover, the SRv6 OAM capabilities, which include ping and traceroute support, enhance network operation by offering robust troubleshooting tools. These features allow for monitoring the status and paths of static SIDs, providing visibility into segment routing operations. The SRv6 static SID configuration also supports integration with dynamic tunnels for Layer 3 VPN services, ensuring service resolution over statically configured SIDs. This integration is crucial for maintaining network service quality and resilience, particularly in environments where multi-hop BGP is not supported, thus ensuring continuity and stability in network operations.

## Binding SID (BSID) for Static SRv6 TE Tunnels

### SUMMARY

The SRv6 binding SID (BSID) helps isolate or decouple different source-routed domains, improving network scalability. The BSID allows the segregation of the network into multiple TE tunnels, each comprising its own stack, backup paths, and compute model. This feature introduces the allocation of SRv6 BSID under SRv6 TE tunnel.

### IN THIS SECTION

- [Benefits of Binding SRv6 SID | 607](#)

With the ingress SRv6-TE feature a service payload is encapsulated in an IPv6 tunnel with an SRH that directs the traffic to take a specific TE path in the network. This SID stack can grow in size thereby increasing the overhead of the packet. Some devices in the network might not support such big stacks in incoming packets. In addition to this, it is impractical to encode the whole stack in the ingress node in inter-domain networks. Encoding the whole stack might result in a very heavy TE database and might also cause convergence issues. If there is a transit MPLS domain, then the labels and SRv6 SIDs might result in too much of an overhead.

The SRv6 binding SID provides a solution to this increasing overhead of a packet. With this feature, you can allocate SRv6 binding SID under an SRv6-TE tunnel. The ingress node only requires a subset of TE database of other domains. The convergence is taken care of at transit nodes that are provisioned with

the binding SID resulting in a comparatively reduced size stack at the ingress node. You can further reduce the stack size with micro SRv6 SID compression.

The SRv6 binding supports both classic and micro SIDs:

- **Classic SRv6 binding SID:** This is a variation of the END behaviour of classic SIDs, where the Endpoint is bound to an SRv6 policy with encapsulation.
  - **END.B6.Encaps**  
This is a non-reduced mode implementation and thus the outer IPv6 encap's destination address is in the SRH.  
  
A transit node configured with the binding SID receives an incoming packet with its destination address matching the END.B6.Encaps through longest prefix match and encapsulates the outgoing packet with an SRH.
  - **END.B6.Encaps.Red**  
Endpoint is bound to an SRv6 policy with reduced SRH encapsulation. There may or may not be an SRH for the outer IPv6 Encap. The outer IPv6 Encap's destination address is not added to the SRH.
- **Micro SRv6 Binding SID**  
This is a variation of the END behaviour of micro SIDS, where the Endpoint is bound to an SRv6 policy with encapsulation.
  - **END.B6.Encaps with Next-C-SID**  
This is a non-reduced mode implementation for micro SIDs and thus the outer IPv6 encap's destination address is in the SRH of the micro SID.
  - **END.B6.Encaps.Red with Next-C-SID** Endpoint is bound to an SRv6 policy with reduced SRH encapsulation for micro SIDs. There may or may not be an SRH for the outer IPv6 Encap. The outer IPv6 Encap's destination address is not added to the SRH of micro SIDs.



**NOTE:** We do not support endpoint behavior flavors like USD, USP or PSP associated with an SRv6 BSID. Endpoint behaviors such as Insert mode SRv6 Binding SID endpoint behaviors are not supported.

Only default lengths are supported for END.B6.Encaps with Next-C-SID and END.B6.Encaps.Red with Next-C-SID.

To enable transit functionality of a static SRv6 TE tunnel through SRv6 Binding SID:

- For classic SIDS, include the `srv6-binding-sid` statement at the [edit protocols source-packet-routing source-routing-path] hierarchy level.

- For micro SIDS, the `micro-srv6-binding-sid` statement at the `[edit protocols source-packet-routing source-routing-path]` hierarchy level.

## Benefits of Binding SRv6 SID

- The Ingress node requires only a subset of TE database of other domains.
- Convergence is taken care of at each transit nodes provisioned with Binding SID
- A cross-data plane switch (SRv6 to MPLS) can be achieved. From ingress the packet can only have SRv6 SIDs. Each transit where an SRv6 Binding SID is provisioned can do the data plane switch depending on the type of Binding SID provisioned.

## RELATED DOCUMENTATION

*srv6-binding-sid*

*micro-srv6-binding-sid*

## How to Enable Link Delay Measurement and Advertising in IGP

### IN THIS SECTION

- [Understanding Link Delay Measurement and Advertising in IGP | 607](#)
- [Example: Enable IS-IS Link Delay with Source Packet Routing in Networking \(SPRING\) in a Layer 3 Virtual Private Network \(VPN\) | 609](#)
- [Configuring OSPF Link Delay and Delay Normalization on an OSPF Interface | 658](#)

## Understanding Link Delay Measurement and Advertising in IGP

### IN THIS SECTION

- [Benefits of link delay measurement and advertising in IS-IS | 608](#)
- [Overview of link delay measurement and advertising in IGP | 608](#)

## Benefits of link delay measurement and advertising in IS-IS

Link delay measurement and advertising in IS-IS provides the following benefits:

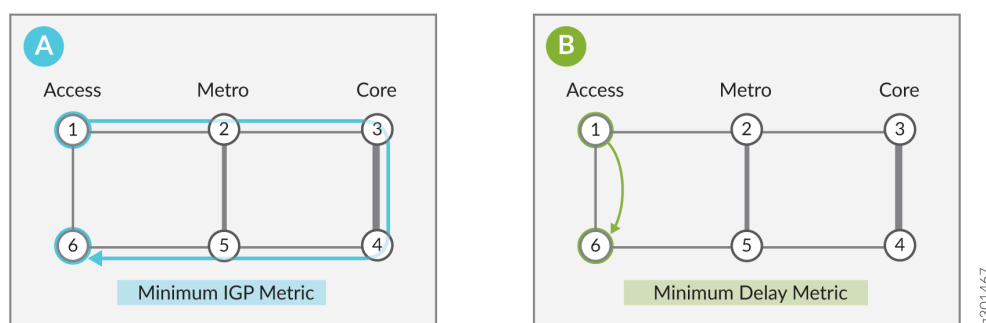
- Highly beneficial in certain networks such as stock market data providers, where it is crucial to have access to market data in real-time to make trades faster than the competition. This is where network performance criteria or latency is becoming critical to data-path selection.
- Helps to make path-selection decisions based on performance data (such as latency) in a cost-effective and scalable way.
- Superior alternative to using metrics such as hop count or cost as routing metrics.

## Overview of link delay measurement and advertising in IGP

Network performance is measured by using TWAMP -Light. You can get the measurement of various performance metrics in IP networks, by using probe messages. IS-IS Traffic Engineering Extensions helps to distribute network-performance information in a scalable fashion. This information can then be used to make path-selection decisions based on network performance.

Border Gateway Protocol Link-State (BGP-LS) allows BGP to carry link-state information acquired from IGPs, which then allows internet service providers (ISP) to selectively expose the information with other ISPs, service providers, CDNs and so on, through normal BGP peering. New BGP-Link State (BGP-LS) TLVs are defined to carry the IGP Traffic Engineering Metric Extensions.

The following illustration depicts the minimum IGP metric and minimum delay metric in networks that consist a core, metro, and access network.



In this scenario, core network is cheaper but has longer delay. Access shortcut, with lowest latency is expensive. As core network is cheaper, majority of traffic typically go from 1>2>3>4>5> to 6 by using

minimum IGP metric. As displayed in scenario a), you can achieve minimum IGP requirement by running IS-IS with appropriate cost configured and default IS-IS algorithm set to zero. In businesses where ultra-low latency is crucial, packets need to go from 1 to 6. As displayed in scenario b), you can achieve minimum delay metric by defining IS-IS flex algorithm with minimum latency, which minimize the delay to the endpoint. This flex algorithm consists only node 1 and node 6.

## Overview of Link Delay Normalization

The Delay Normalization feature computes a normalized delay value and uses the normalized value instead of measured delay value. This value is advertised and used as a metric during the Flexible Algorithm computation. The normalization is performed when the delay is received from the delay measurement component. When the next value is received, it is normalized and compared to the previously saved normalized value. If the values are different, then the LSP generation is triggered.

Delay normalization is disabled by default. To enable and configure delay normalization across IGP instances, use the delay-measurement command.

```
set protocols isis interface interface-name delay-measurement normalize interval value offset value
```

```
set protocols ospf area aread interface interface-name delay-measurement normalize interval value offset value
```

## SEE ALSO

[delay-measurement \(Protocols IS-IS\)](#)

[delay-measurement \(Protocols OSPF\)](#)

[delay-metric \(Protocols OSPF\)](#)

[Understand Two-Way Active Measurement Protocol](#)

## Example: Enable IS-IS Link Delay with Source Packet Routing in Networking (SPRING) in a Layer 3 Virtual Private Network (VPN)

### IN THIS SECTION

 [Requirements](#) | 610

- [Overview | 612](#)
- [Configuration | 613](#)
- [Verification | 643](#)

This example shows how to configure IS-IS link delay with SPRING in a Layer3 VPN scenario. In the example, you can create two VPNs between PE1 and PE2. VPN1 optimizes link delay and VPN2 optimizes IGP metric. Although you can configure the feature to enable bidirectional traffic in the test topology, we're focusing on a unidirectional traffic scenario in this example. Specifically, your task is to control the forwarding path for Layer 3 VPN traffic sent by PE1 to destinations advertised by PE2.

## Requirements

### IN THIS SECTION

- [Topology | 611](#)

This example uses the following hardware and software components:

- Four MX Series routers
- Junos OS Release 21.1R1 or later running on all devices





traffic using flex algorithm 129 has two equal cost paths between PE1 and PE2, both incurring two hops and a resulting metric of 20.

## Overview

In IP networks, the bulk of traffic often goes through the core network, which reduces costs but might result in increased latency. Business traffic, however, often benefits from the ability to make path-selection decisions based on other performance metrics, such as path latency, rather than relaying on the traditional path optimization based simply on IGP metrics. Optimizing a path to reduce latency can greatly benefit applications like real-time voice and video. It can also enable high performance access to financial market data where milliseconds can translate into significant gains or losses.

You can enable IS-IS link delay in IP networks. You can achieve minimum IGP metric paths by configuring IS-IS with the appropriate link cost using the default IS-IS algorithm (0). Doing so optimizes paths to the endpoint that are based strictly on the sum of the link metrics. By using the IS-IS delay flex algorithm you can optimize paths based on their end-to-end delay.

Link delay can be dynamically measured using Two-Way Active Measurement Probes (TWAMP). The routers then flood their link delay parameters. The routers in the area store these parameters in the shared Link State Database (LSDB). Ingress nodes run an SPF algorithm against the LSDB to compute paths that are optimized on various attributes, such as link colors, IGP metric, traffic-engineering (TE) metric, or as shown in this example, link delay.



### NOTE:

1. Delay of link is measured in microseconds level. In one notification it may be 1000 microseconds, and in next notification it may be 4000 microseconds. In terms of microseconds it is 400 percent change. In terms of milliseconds it is a huge change but in terms of milliseconds it is not an appreciable change. Therefore, as per the requirement, you can set the optimal value otherwise every few minutes new delay will be advertised which can lead to frequent SPF computation and routes computation.
2. During periodic announcement interval (default 120 seconds) check if max Unidirectional Link Delay measured parameter crosses periodic threshold then report the same to registered modules.

*set protocols isis interface <interface\_name> delay-measurement advertisement periodic threshold < 0-100>.* The default threshold value is 10 percent and the value ranges from 0 to 100.

The egress router signals which flex algorithm is desired by attaching an associated color community to routes advertised through BGP. At the sending end (the local PE that has received the tagged routes advertised by the remote PE), these color communities are used to index into a color table that resolves

the remote protocol next hop (the PE's loopback address) to a flex algorithm identifier. In the context of Layer 3 VPNs a color mapping policy is used at the ingress node to select which prefixes should have their next hops resolved via the color table.

The local PE then uses its local Flex Algorithm Definition (FAD) to map the flex algorithm identifier into a set of path selection criteria, for example "use blue links and optimize on delay". The ingress PE calculates the optimal path based on the values in the LSDB, pushes the related MPLS label stack onto the packet, and sends it to the associated next hop. This results in traffic-engineered MPLS paths using IS-IS as the signaling protocol.

## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 613](#)
- [Step-by-step Procedure | 624](#)
- [Results | 636](#)

### *CLI Quick Configuration*

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.



**NOTE:** Depending on the type of MPCs in your MX Series routers you might need to explicitly enable enhanced IP services to support the IS-IS delay feature. When you commit the `set chassis network-services enhanced-ip` configuration statement, you will be prompted to reboot the system.

### PE1

```
set system host-name PE1
set chassis network-services enhanced-ip
set services rpm twamp server authentication-mode none
set services rpm twamp server light
set interfaces ge-0/0/0 description To_R1
set interfaces ge-0/0/0 unit 0 family inet address 10.0.1.10/24
set interfaces ge-0/0/0 unit 0 family iso
```

```

set interfaces ge-0/0/0 unit 0 family inet6 address 2001:db8:10:0:1::10/80
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 16
set interfaces ge-0/0/1 description To_R2
set interfaces ge-0/0/1 unit 0 family inet address 10.0.2.10/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family inet6 address 2001:db8:10:0:2::10/80
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 16
set interfaces lo0 unit 0 family inet address 192.168.255.10/32
set interfaces lo0 unit 0 family inet address 127.0.0.1/32
set interfaces lo0 unit 0 family iso address 49.0001.000a.0a0a.0a00
set interfaces lo0 unit 0 family inet6 address 2001:db8:192:168:255::10/128
set interfaces lo0 unit 1 family inet address 172.16.10.1/32
set interfaces lo0 unit 1 family inet6 address 2001:db8:172:16:10::1/128
set interfaces lo0 unit 2 family inet address 172.16.10.2/32
set interfaces lo0 unit 2 family inet6 address 2001:db8:172:16:10::2/128
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 192.168.255.10/32 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128 index
1280
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128 node-
segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129 index
1290
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129 node-
segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1000
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept
set policy-options policy-statement prefix-sid term 2 from route-filter
2001:db8:192:168:255::10/128 exact
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128 index
4280
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128 node-
segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129 index
4290
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129 node-
segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment index 4000
set policy-options policy-statement prefix-sid term 2 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 2 then accept
set policy-options policy-statement v6vpn1_res_map1 from route-filter 2001:db8:172:16:1::/80
orlonger

```

```

set policy-options policy-statement v6vpn1_res_map1 then accept
set policy-options policy-statement v6vpn1_res_map1 then resolution-map map1
set policy-options policy-statement v6vpn2_res_map1 from route-filter 2001:db8:172:16:2::/80
orlonger
set policy-options policy-statement v6vpn2_res_map1 then accept
set policy-options policy-statement v6vpn2_res_map1 then resolution-map map1
set policy-options policy-statement vpn1_res_map1 term 1 from route-filter 172.16.1.0/24 orlonger
set policy-options policy-statement vpn1_res_map1 term 1 then accept
set policy-options policy-statement vpn1_res_map1 term 1 then resolution-map map1
set policy-options policy-statement vpn2_res_map1 term 1 from route-filter 172.16.2.0/24 orlonger
set policy-options policy-statement vpn2_res_map1 term 1 then accept
set policy-options policy-statement vpn2_res_map1 term 1 then resolution-map map1
set policy-options resolution-map map1 mode ip-color
set routing-instances vpn1 instance-type vrf
set routing-instances vpn1 interface lo0.1
set routing-instances vpn1 route-distinguisher 64512:1
set routing-instances vpn1 vrf-target target:64512:1
set routing-instances vpn1 vrf-table-label
set routing-instances vpn2 instance-type vrf
set routing-instances vpn2 interface lo0.2
set routing-instances vpn2 route-distinguisher 64512:2
set routing-instances vpn2 vrf-target target:64512:2
set routing-instances vpn2 vrf-table-label
set protocols bgp group to-RRv6 type internal
set protocols bgp group to-RRv6 local-address 2001:db8:192:168:255::10
set protocols bgp group to-RRv6 import v6vpn1_res_map1
set protocols bgp group to-RRv6 import v6vpn2_res_map1
set protocols bgp group to-RRv6 family inet6 unicast extended-nexthop-color
set protocols bgp group to-RRv6 family inet6-vpn unicast
set protocols bgp group to-RRv6 neighbor 2001:db8:192:168:255::2
set protocols bgp group to-RR type internal
set protocols bgp group to-RR local-address 192.168.255.10
set protocols bgp group to-RR import vpn1_res_map1
set protocols bgp group to-RR import vpn2_res_map1
set protocols bgp group to-RR family inet unicast extended-nexthop-color
set protocols bgp group to-RR family inet-vpn unicast
set protocols bgp group to-RR family traffic-engineering unicast
set protocols bgp group to-RR neighbor 192.168.255.2
set protocols bgp group to-RR vpn-apply-export
set protocols isis interface ge-0/0/0.0 level 2 post-convergence-lfa node-protection
set protocols isis interface ge-0/0/0.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 post-convergence-lfa node-protection

```

```

set protocols isis interface ge-0/0/1.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 80000
set protocols isis source-packet-routing srgb index-range 5000
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis source-packet-routing flex-algorithm 129
set protocols isis level 1 disable
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis traffic-engineering l3-unicast-topology
set protocols isis traffic-engineering advertisement always
set protocols isis export prefix-sid
set protocols mpls traffic-engineering
set protocols mpls admin-groups RED 0
set protocols mpls admin-groups BLUE 1
set protocols mpls icmp-tunneling
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols mpls interface ge-0/0/0.0 admin-group RED
set protocols mpls interface ge-0/0/1.0 admin-group BLUE
set routing-options flex-algorithm 128 definition metric-type delay-metric
set routing-options flex-algorithm 128 definition spf
set routing-options flex-algorithm 128 definition admin-group include-any BLUE
set routing-options flex-algorithm 129 definition metric-type igp-metric
set routing-options flex-algorithm 129 definition spf
set routing-options flex-algorithm 129 definition admin-group include-any RED
set routing-options flex-algorithm 129 definition admin-group include-any BLUE
set routing-options router-id 192.168.255.10
set routing-options autonomous-system 64512
set routing-options forwarding-table export pplb
set routing-options forwarding-table ecmp-fast-reroute
set routing-options forwarding-table chained-composite-next-hop ingress l3vpn

```

P1

```

set system host-name P1
set chassis network-services enhanced-ip
set services rpm twamp server authentication-mode none
set services rpm twamp server light
set interfaces ge-0/0/0 description To_R0
set interfaces ge-0/0/0 unit 0 family inet address 10.0.1.1/24

```

```

set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family inet6 address 2001:db8:10:0:1::1/80
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 16
set interfaces ge-0/0/1 description To_R2
set interfaces ge-0/0/1 unit 0 family inet address 10.0.12.1/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family inet6 address 2001:db8:10:0:12::1/80
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 16
set interfaces ge-0/0/2 description To_R3
set interfaces ge-0/0/2 unit 0 family inet address 10.0.13.1/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/2 unit 0 family inet6 address 2001:db8:10:0:13::1/80
set interfaces ge-0/0/2 unit 0 family mpls maximum-labels 16
set interfaces lo0 unit 0 family inet address 192.168.255.1/32
set interfaces lo0 unit 0 family iso address 49.0001.0005.0505.0500
set interfaces lo0 unit 0 family inet6 address 2001:db8:192:168:255::1/128
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 192.168.255.1/32 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128 index
1281
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128 node-
segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129 index
1291
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129 node-
segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1001
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept
set policy-options policy-statement prefix-sid term 2 from route-filter
2001:db8:192:168:255::1/128 exact
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128 index
4281
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128 node-
segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129 index
4291
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129 node-
segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment index 4001
set policy-options policy-statement prefix-sid term 2 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 2 then accept
set protocols isis interface ge-0/0/0.0 level 2 post-convergence-lfa node-protection

```

```

set protocols isis interface ge-0/0/0.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 post-convergence-lfa node-protection
set protocols isis interface ge-0/0/1.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/2.0 level 2 post-convergence-lfa node-protection
set protocols isis interface ge-0/0/2.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 80000
set protocols isis source-packet-routing srgb index-range 5000
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis source-packet-routing flex-algorithm 129
set protocols isis level 1 disable
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis traffic-engineering l3-unicast-topology
set protocols isis traffic-engineering advertisement always
set protocols isis export prefix-sid
set protocols mpls admin-groups RED 0
set protocols mpls admin-groups BLUE 1
set protocols mpls icmp-tunneling      set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols mpls interface ge-0/0/0.0 admin-group RED
set protocols mpls interface ge-0/0/1.0 admin-group BLUE
set protocols mpls interface ge-0/0/2.0 admin-group BLUE
set routing-options router-id 192.168.255.1
set routing-options autonomous-system 65412
set routing-options forwarding-table export pplb

```

## P2

```

set system host-name P2
set chassis network-services enhanced-ip
set services rpm twamp server authentication-mode none
set services rpm twamp server light
set interfaces ge-0/0/0 unit 0 family inet address 10.0.2.2/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family inet6 address 2001:db8:10:0:2::2/80
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 16
set interfaces ge-0/0/1 description To_R1
set interfaces ge-0/0/1 unit 0 family inet address 10.0.12.2/24

```



```

set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family inet6 address 2001:db8:10:0:12::2/80
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 16
set interfaces ge-0/0/2 description To_R3
set interfaces ge-0/0/2 unit 0 family inet address 10.0.23.2/24
set interfaces ge-0/0/2 unit 0 family iso
set interfaces ge-0/0/2 unit 0 family inet6 address 2001:db8:10:0:23::2/80
set interfaces ge-0/0/2 unit 0 family mpls maximum-labels 16
set interfaces lo0 unit 0 family inet address 192.168.255.2/32
set interfaces lo0 unit 0 family iso address 49.0001.0002.0202.0200
set interfaces lo0 unit 0 family inet6 address 2001:db8:192:168:255::2/128
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 192.168.255.2/32 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128 index
1282
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128 node-
segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129 index
1292
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129 node-
segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1002
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept
set policy-options policy-statement prefix-sid term 2 from route-filter
2001:db8:192:168:255::2/128 exact
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128 index
4282
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128 node-
segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129 index
4292
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129 node-
segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment index 4002
set policy-options policy-statement prefix-sid term 2 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 2 then accept
set policy-options policy-statement ted2nlri_igp term 1 from family traffic-engineering
set policy-options policy-statement ted2nlri_igp term 1 from protocol isis
set policy-options policy-statement ted2nlri_igp term 1 then accept
set protocols bgp group to-RRv6 type internal
set protocols bgp group to-RRv6 local-address 2001:db8:192:168:255::2
set protocols bgp group to-RRv6 family inet6 unicast

```

```

set protocols bgp group to-RRv6 family inet6-vpn unicast
set protocols bgp group to-RRv6 neighbor 2001:db8:192:168:255::10
set protocols bgp group to-RRv6 neighbor 2001:db8:192:168:255::3
set protocols bgp group to-RR type internal
set protocols bgp group to-RR local-address 192.168.255.2
set protocols bgp group to-RR family inet unicast
set protocols bgp group to-RR family inet-vpn unicast
set protocols bgp group to-RR neighbor 192.168.255.10
set protocols bgp group to-RR neighbor 192.168.255.3
set protocols bgp cluster 192.168.255.2
set protocols isis interface ge-0/0/0.0 level 2 post-convergence-lfa node-protection
set protocols isis interface ge-0/0/0.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 post-convergence-lfa node-protection
set protocols isis interface ge-0/0/1.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/2.0 level 2 post-convergence-lfa node-protection
set protocols isis interface ge-0/0/2.0 delay-metric 20000
set protocols isis interface ge-0/0/2.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/2.0 point-to-point
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 80000
set protocols isis source-packet-routing srgb index-range 5000
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis source-packet-routing flex-algorithm 129
set protocols isis level 1 disable
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis traffic-engineering l3-unicast-topology
set protocols isis traffic-engineering advertisement always
set protocols isis export prefix-sid
set protocols mpls traffic-engineering
set protocols mpls admin-groups RED 0
set protocols mpls admin-groups BLUE 1
set protocols mpls icmp-tunneling          set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols mpls interface ge-0/0/0.0 admin-group BLUE
set protocols mpls interface ge-0/0/1.0 admin-group BLUE
set protocols mpls interface ge-0/0/2.0 admin-group BLUE
set routing-options router-id 192.168.255.2
set routing-options autonomous-system 64512
set routing-options forwarding-table export pplb

```

## PE2

```

set system host-name PE2
set chassis network-services enhanced-ip
set services rpm twamp server authentication-mode none
set services rpm twamp server light
set interfaces ge-0/0/0 description To_R1
set interfaces ge-0/0/0 unit 0 family inet address 10.0.13.3/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family inet6 address 2001:db8:10:0:13::3/80
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 16
set interfaces ge-0/0/1 description To_R2
set interfaces ge-0/0/1 unit 0 family inet address 10.0.23.3/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family inet6 address 2001:db8:10:0:23::364/128
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 16
set interfaces lo0 unit 0 family inet address 192.168.255.3/32
set interfaces lo0 unit 0 family inet address 127.0.0.1/32
set interfaces lo0 unit 0 family iso address 49.0001.0007.0707.0700
set interfaces lo0 unit 0 family inet6 address 2001:db8:192:168:255::3/128
set interfaces lo0 unit 1 family inet address 172.16.3.1/32
set interfaces lo0 unit 1 family inet6 address 2001:db8:172:16:3::1/128
set interfaces lo0 unit 2 family inet address 172.16.3.2/32
set interfaces lo0 unit 2 family inet6 address 2001:db8:172:16:3::2/128
set policy-options policy-statement pplb then load-balance per-packet
set policy-options policy-statement prefix-sid term 1 from route-filter 192.168.255.3/32 exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128 index
1283
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128 node-
segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129 index
1293
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129 node-
segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1003
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept
set policy-options policy-statement prefix-sid term 2 from route-filter
2001:db8:192:168:255::3/128 exact
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128 index
4283
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128 node-

```

```

segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129 index
4293
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129 node-
segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment index 4003
set policy-options policy-statement prefix-sid term 2 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 2 then accept
set policy-options policy-statement vpn_1_export term 1 from route-filter 172.16.1.0/24 orlonger
set policy-options policy-statement vpn_1_export term 1 then community add color128
set policy-options policy-statement vpn_1_export term 1 then next-hop 192.168.255.3
set policy-options policy-statement vpn_1_export term 1 then accept
set policy-options policy-statement vpn_1_export_v6 term 1 from route-filter
2001:db8:172:16:1::/80 orlonger
set policy-options policy-statement vpn_1_export_v6 term 1 then community add color128
set policy-options policy-statement vpn_1_export_v6 term 1 then next-hop 2001:db8:192:168:255::3
set policy-options policy-statement vpn_1_export_v6 term 1 then accept
set policy-options policy-statement vpn_1_export_v6 term 2 from route-filter
2001:db8:172:16:3::1/128 exact
set policy-options policy-statement vpn_1_export_v6 term 2 then community add color128
set policy-options policy-statement vpn_1_export_v6 term 2 then next-hop 2001:db8:192:168:255::3
set policy-options policy-statement vpn_1_export_v6 term 2 then accept
set policy-options policy-statement vpn_2_export term 1 from route-filter 172.16.2.0/24 orlonger
set policy-options policy-statement vpn_2_export term 1 then community add color129
set policy-options policy-statement vpn_2_export term 1 then next-hop 192.168.255.3
set policy-options policy-statement vpn_2_export term 1 then accept
set policy-options policy-statement vpn_2_export_v6 term 1 from route-filter
2001:db8:172:16:2::/80 orlonger
set policy-options policy-statement vpn_2_export_v6 term 1 then community add color129
set policy-options policy-statement vpn_2_export_v6 term 1 then next-hop 2001:db8:192:168:255::3
set policy-options policy-statement vpn_2_export_v6 term 1 then accept
set policy-options community color128 members color:0:128
set policy-options community color129 members color:0:129
set policy-options resolution-map map1 mode ip-color
set routing-instances vpn1 instance-type vrf
set routing-instances vpn1 routing-options rib vpn1.inet6.0 static route 2001:db8:172:16:1::/80
receive
set routing-instances vpn1 routing-options static route 172.16.1.0/24 receive
set routing-instances vpn1 interface lo0.1
set routing-instances vpn1 route-distinguisher 64512:1
set routing-instances vpn1 vrf-target target:64512:1
set routing-instances vpn1 vrf-table-label
set routing-instances vpn2 instance-type vrf

```

```

set routing-instances vpn2 routing-options rib vpn2.inet6.0 static route 2001:db8:172:16:2::/80
receive
set routing-instances vpn2 routing-options static route 172.16.2.0/24 receive
set routing-instances vpn2 interface lo0.2
set routing-instances vpn2 route-distinguisher 64512:2
set routing-instances vpn2 vrf-target target:64512:2
set routing-instances vpn2 vrf-table-label
set protocols bgp group to-RRv6 type internal
set protocols bgp group to-RRv6 local-address 2001:db8:192:168:255::3
set protocols bgp group to-RRv6 family inet6 unicast extended-nexthop-color
set protocols bgp group to-RRv6 family inet6-vpn unicast
set protocols bgp group to-RRv6 export vpn_1_export_v6
set protocols bgp group to-RRv6 export vpn_2_export_v6
set protocols bgp group to-RRv6 neighbor 2001:db8:192:168:255::2
set protocols bgp group to-RRv6 vpn-apply-export
set protocols bgp group to-RR type internal
set protocols bgp group to-RR local-address 192.168.255.3
set protocols bgp group to-RR family inet unicast extended-nexthop-color
set protocols bgp group to-RR family inet-vpn unicast
set protocols bgp group to-RR export vpn_1_export
set protocols bgp group to-RR export vpn_2_export
set protocols bgp group to-RR neighbor 192.168.255.2
set protocols bgp group to-RR vpn-apply-export
set protocols isis interface ge-0/0/0.0 level 2 post-convergence-lfa node-protection
set protocols isis interface ge-0/0/0.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 post-convergence-lfa node-protection
set protocols isis interface ge-0/0/1.0 delay-measurement advertisement periodic threshold 100
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/2.0 delay-metric 20000
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 80000
set protocols isis source-packet-routing srgb index-range 5000
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis source-packet-routing flex-algorithm 129
set protocols isis level 1 disable
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis traffic-engineering l3-unicast-topology
set protocols isis traffic-engineering advertisement always
set protocols isis export prefix-sid
set protocols mpls admin-groups RED 0
set protocols mpls admin-groups BLUE 1

```

```

set protocols mpls icmp-tunneling      set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols mpls interface ge-0/0/0.0 admin-group BLUE
set protocols mpls interface ge-0/0/1.0 admin-group BLUE
set routing-options router-id 192.168.255.3
set routing-options autonomous-system 64512
set routing-options forwarding-table export pplb
set routing-options forwarding-table ecmp-fast-reroute
set routing-options forwarding-table chained-composite-next-hop ingress l3vpn

```

### *Step-by-step Procedure*

1. Configure the basic device settings such as hostname, IPv4, IPv6 addresses, loopback interface addresses, enhanced-ip mode, and enable the ISO and MPLS protocol families on all interfaces of all 4 routers.

```

user@PE1#
set system host-name PE1
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 description To_R1
set interfaces ge-0/0/0 unit 0 family inet address 10.0.1.10/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family inet6 address 2001:db8:10:0:1::10/80
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 16
set interfaces ge-0/0/1 description To_R2
set interfaces ge-0/0/1 unit 0 family inet address 10.0.2.10/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family inet6 address 2001:db8:10:0:2::10/80
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 16
set interfaces lo0 unit 0 family inet address 192.168.255.10/32
set interfaces lo0 unit 0 family inet address 127.0.0.1/32
set interfaces lo0 unit 0 family iso address 49.0001.000a.0a0a.0a00
set interfaces lo0 unit 0 family inet6 address 2001:db8:192:168:255::10/128
set interfaces lo0 unit 1 family inet address 172.16.10.1/32
set interfaces lo0 unit 1 family inet6 address 2001:db8:172:16:10::1/128
set interfaces lo0 unit 2 family inet address 172.16.10.2/32
set interfaces lo0 unit 2 family inet6 address 2001:db8:172:16:10::2/128

```

2. Configure the router-ID, autonomous system (AS) number, and apply a load balancing export policy to the forwarding table on all routers to enable load balancing of traffic.

```
user@PE1#
set routing-options router-id 192.168.255.10
set routing-options autonomous-system 64512
set routing-options forwarding-table export pplb
```

3. On PE1 and PE2, configure equal-cost multipath (ECMP) to enable fast reroute protection. Also configure chained composite next hop to allow the routers to point routes that share the same destination to a common forwarding next hop. This option improves forwarding information base (FIB) scaling.

```
user@PE1#
set routing-options forwarding-table ecmp-fast-reroute
set routing-options forwarding-table chained-composite-next-hop ingress l3vpn
```

4. Enable MPLS protocol processing on all interfaces at all routers. Also enable traffic engineering.

```
user@PE1#
set protocols mpls interface fxp0.0 disable
set protocols mpls interface all
set protocols mpls traffic-engineering
```

5. Enable TWAMP probes on all routers. These probes support dynamic measurement of the link delay between each pair of routers.

```
user@PE1#
set services rpm twamp server authentication-mode none
set services rpm twamp server light
```

6. Configure the IS-IS protocol for point-to-point operation (TWAMP based delay measurements are not supported on multi-point links), and enable node protection mode for Topology-Independent Loop-Free Alternate (TILFA) operation on all interfaces. You also enable passive mode IS-IS on the loopback interface and disable IS-IS level 1 to use only IS-IS level 2. Enable traffic engineering with layer 3 unicast topology to download IGP topology into the TED. Configure IS-IS to support SPRING routed paths. The *prefix-sid* export policy is defined in a subsequent step. This policy is

used to have the local node advertise its loopback address with a mapping to one or more flex algorithms.

```
user@PE1#
set protocols isis level 1 disable
set protocols isis interface ge-0/0/0.0 point-to-point
set protocols isis interface ge-0/0/0.0 level 2 post-convergence-lfa node-protection
set protocols isis interface ge-0/0/1.0 point-to-point
set protocols isis interface ge-0/0/1.0 level 2 post-convergence-lfa node-protection
set protocols isis interface lo0.0 passive
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis traffic-engineering l3-unicast-topology
set protocols isis traffic-engineering advertisement always
set protocols isis export prefix-sid
```

7. Configure dynamic IS-IS link delay-measurement using TWAMP probes on all IS-IS interfaces at all routers (except for the link between P2 and PE2, which uses a static delay value in this example).

```
user@PE1#
set protocols isis interface ge-0/0/0.0 delay-measurement advertisement periodic threshold
100
set protocols isis interface ge-0/0/1.0 delay-measurement advertisement periodic threshold
100
```

```
user@P1#
set protocols isis interface ge-0/0/0.0 delay-measurement advertisement periodic threshold
100
set protocols isis interface ge-0/0/1.0 delay-measurement advertisement periodic threshold
100
set protocols isis interface ge-0/0/2.0 delay-measurement advertisement periodic threshold
100
```

```
user@P2#
set protocols isis interface ge-0/0/0.0 delay-measurement advertisement periodic threshold
100
set protocols isis interface ge-0/0/1.0 delay-measurement advertisement periodic threshold
100
```



```
set protocols isis interface ge-0/0/2.0 delay-measurement advertisement periodic threshold
100
```

```
user@PE2#
set protocols isis interface ge-0/0/0.0 delay-measurement advertisement periodic threshold
100
set protocols isis interface ge-0/0/1.0 delay-measurement advertisement periodic threshold
100
```

8. Configure the static delay-metric on the link between P2 and PE2.

```
user@P2#
set protocols isis interface ge-0/0/2.0 delay-metric 20000
```

```
user@PE2#
set protocols isis interface ge-0/0/1.0 delay-metric 20000
```

9. Configure PE1 and PE2 to support two Layer 3 VPNs (VPN1 and VPN2).

```
user@PE1#
set routing-instances vpn1 instance-type vrf
set routing-instances vpn1 interface lo0.1
set routing-instances vpn1 route-distinguisher 64512:1
set routing-instances vpn1 vrf-target target:64512:1
set routing-instances vpn1 vrf-table-label

set routing-instances vpn2 instance-type vrf
set routing-instances vpn2 interface lo0.2
set routing-instances vpn2 route-distinguisher 64512:2
set routing-instances vpn2 vrf-target target:64512:2
set routing-instances vpn2 vrf-table-label
```



**NOTE:** Note that the routing instances at PE2 are configured with IPv4 and IPv6 static routes. These routes are configured with the `receive` option to allow you to test connectivity using ping. The IS-IS delay feature operates the same if the Layer 3 VPN uses a dynamic routing protocol between the PE and an attached CE device.

We use static routes in this example to keep the topology simple to allow focus on the IS-IS delay optimization feature.

```

user@PE2#
set routing-instances vpn1 instance-type vrf
set routing-instances vpn1 routing-options rib vpn1.inet6.0 static route
2001:db8:172:16:1::/80 receive
set routing-instances vpn1 routing-options static route 172.16.1.0/24 receive
set routing-instances vpn1 interface lo0.1
set routing-instances vpn1 route-distinguisher 64512:1
set routing-instances vpn1 vrf-target target:64512:1
set routing-instances vpn1 vrf-table-label

set routing-instances vpn2 instance-type vrf
set routing-instances vpn2 routing-options rib vpn2.inet6.0 static route
2001:db8:172:16:2::/80 receive
set routing-instances vpn2 routing-options static route 172.16.2.0/24 receive
set routing-instances vpn2 interface lo0.2
set routing-instances vpn2 route-distinguisher 64512:2
set routing-instances vpn2 vrf-target target:64512:2
set routing-instances vpn2 vrf-table-label

```

10. Configure a map policy at PE1 to enable VPN route resolution for matching prefixes against the BGP color table. This allows you to evoke flex path forwarding algorithms on a per-prefix basis. The *map1* resolution policy is set to the ip-color resolution mode.



**NOTE:** In a Layer 3 VPN context a mapping policy is needed to select which prefixes are allowed to have their next hop resolved in the color table. Simply having routes with extended next hops and color communities attached does not result in the use of the color table, unless a mapping policy is used.

```

user@PE1#
set policy-options policy-statement vpn1_res_map1 term 1 from route-filter 172.16.1.0/24
orlonger
set policy-options policy-statement vpn1_res_map1 term 1 then accept
set policy-options policy-statement vpn1_res_map1 term 1 then resolution-map map1
set policy-options policy-statement vpn2_res_map1 term 1 from route-filter 172.16.2.0/24
orlonger
set policy-options policy-statement vpn2_res_map1 term 1 then accept

```

```

set policy-options policy-statement vpn2_res_map1 term 1 then resolution-map map1
set policy-options policy-statement v6vpn1_res_map1 from route-filter
2001:db8:172:16:1::/80 orlonger
set policy-options policy-statement v6vpn1_res_map1 then accept
set policy-options policy-statement v6vpn1_res_map1 then resolution-map map1
set policy-options policy-statement v6vpn2_res_map1 from route-filter
2001:db8:172:16:2::/80 orlonger
set policy-options policy-statement v6vpn2_res_map1 then accept
set policy-options policy-statement v6vpn2_res_map1 then resolution-map map1
set policy-options resolution-map map1 mode ip-color

```

11. Configure VPN route export policies at PE2 to attach the desired color communities to the VPN routes it advertises to PE1 (via the route reflector). Of significance here is how the routes from VPN1 have the color community for flex path 128 (optimize delay) attached, while the routes advertised from VPN2 have the 129 color community attached (optimize IGP metric).

```

user@PE2#
set policy-options policy-statement vpn_1_export term 1 from route-filter 172.16.1.0/24
orlonger
set policy-options policy-statement vpn_1_export term 1 then community add color128
set policy-options policy-statement vpn_1_export term 1 then next-hop 192.168.255.3
set policy-options policy-statement vpn_1_export term 1 then accept

set policy-options policy-statement vpn_2_export term 1 from route-filter 172.16.2.0/24
orlonger
set policy-options policy-statement vpn_2_export term 1 then community add color129
set policy-options policy-statement vpn_2_export term 1 then next-hop 192.168.255.3
set policy-options policy-statement vpn_2_export term 1 then accept

set policy-options policy-statement vpn_1_export_v6 term 1 from route-filter
2001:db8:172:16:1::/80 orlonger
set policy-options policy-statement vpn_1_export_v6 term 1 then community add color128
set policy-options policy-statement vpn_1_export_v6 term 1 then next-hop
2001:db8:192:168:255::3
set policy-options policy-statement vpn_1_export_v6 term 1 then accept
set policy-options policy-statement vpn_2_export_v6 term 1 from route-filter
2001:db8:172:16:2::/80 orlonger
set policy-options policy-statement vpn_2_export_v6 term 1 then community add color129
set policy-options policy-statement vpn_2_export_v6 term 1 then next-hop
2001:db8:192:168:255::3
set policy-options policy-statement vpn_2_export_v6 term 1 then accept

```

```
set policy-options community color128 members color:0:128
set policy-options community color129 members color:0:129
```

12. Configure BGP peering between the PE devices and the route reflector. Configure the unicast network layer reachability information (NLRI) to support extended color next hops on the PE devices. Enabling this option allows routes with color communities to have their next hop resolve through the color table. Without the extended next hop setting route with color communities undergoing normal next hop resolution and will not use flex algorithm paths.
13. You also enable support for IPv4 and IPv6 Layer 3 VPN unicast routes. On PE1 you apply the color mapping policies as import, so it can act on the routes received from the remote PE device.

```
user@PE1#
set protocols bgp group to-RR type internal
set protocols bgp group to-RR local-address 192.168.255.10
set protocols bgp group to-RR neighbor 192.168.255.2
set protocols bgp group to-RR family inet unicast extended-nexthop-color
set protocols bgp group to-RR family inet-vpn unicast
set protocols bgp group to-RR family traffic-engineering unicast
set protocols bgp group to-RR import vpn1_res_map1
set protocols bgp group to-RR import vpn2_res_map1
set protocols bgp group to-RRv6 type internal
set protocols bgp group to-RRv6 local-address 2001:db8:192:168:255::10
set protocols bgp group to-RRv6 neighbor 2001:db8:192:168:255::2
set protocols bgp group to-RRv6 family inet6 unicast extended-nexthop-color
set protocols bgp group to-RRv6 family inet6-vpn unicast
set protocols bgp group to-RRv6 import v6vpn1_res_map1
set protocols bgp group to-RRv6 import v6vpn2_res_map1
```

```
user@P2#
set protocols bgp group to-RR type internal
set protocols bgp group to-RR local-address 192.168.255.2
set protocols bgp group to-RR neighbor 192.168.255.10
set protocols bgp group to-RR neighbor 192.168.255.3
set protocols bgp cluster 192.168.255.2
set protocols bgp group to-RR family inet unicast
set protocols bgp group to-RR family inet-vpn unicast
```

On PE 2 you apply export policy to attach the desired color community to the VPN route advertisements sent to PE1. The `vpn-apply-export` option is needed at PE2 to allow the export policies to act on VPN routes advertised to remote PEs.

```
user@PE2#
set protocols bgp group to-RR type internal
set protocols bgp group to-RR local-address 192.168.255.3
set protocols bgp group to-RR neighbor 192.168.255.2
set protocols bgp group to-RR family inet unicast extended-nexthop-color
set protocols bgp group to-RR family inet-vpn unicast
set protocols bgp group to-RR export vpn_1_export
set protocols bgp group to-RR export vpn_2_export
set protocols bgp group to-RR vpn-apply-export

set protocols bgp group to-RRv6 type internal
set protocols bgp group to-RRv6 local-address 2001:db8:192:168:255::3
set protocols bgp group to-RRv6 neighbor 2001:db8:192:168:255::2
set protocols bgp group to-RRv6 family inet6 unicast extended-nexthop-color
set protocols bgp group to-RRv6 family inet6-vpn unicast
set protocols bgp group to-RRv6 export vpn_1_export_v6
set protocols bgp group to-RRv6 export vpn_2_export_v6
set protocols bgp group to-RRv6 vpn-apply-export
```

14. Define the per-packet load balancing policy on all routers.

```
user@PE1#
set policy-options policy-statement pplb then load-balance per-packet
```

15. Configure support for segment routing with two flex algorithms (128 and 129) on all routers.

```
user@PE1#
set protocols isis source-packet-routing srgb start-label 80000
set protocols isis source-packet-routing srgb index-range 5000
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis source-packet-routing flex-algorithm 129
```

16. Configure all routers to advertise their loopback address with support for both the 128 and 129 flex algorithms. The `prefix-segment index` option sets the base label for each router's loopback

address. In this example the IPv4 base index and IPv6 base index is set to reflect the router number. As a result R0 (PE1) uses 1000 for IPv4 while R1 (P1) uses 1001.

```

user@PE1#
set policy-options policy-statement prefix-sid term 1 from route-filter 192.168.255.10/32
exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128
index 1280
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128
node-segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129
index 1290
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129
node-segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1000
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

set policy-options policy-statement prefix-sid term 2 from route-filter
2001:db8:192:168:255::10/128 exact
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128
index 4280
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128
node-segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129
index 4290
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129
node-segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment index 4000
set policy-options policy-statement prefix-sid term 2 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 2 then accept

```

```

user@P1#
set policy-options policy-statement prefix-sid term 1 from route-filter 192.168.255.1/32
exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128
index 1281
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128
node-segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129
index 1291

```

```

set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129
node-segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1001
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

set policy-options policy-statement prefix-sid term 2 from route-filter
2001:db8:192:168:255::1/128 exact
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128
index 4281
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128
node-segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129
index 4291
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129
node-segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment index 4001
set policy-options policy-statement prefix-sid term 2 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 2 then accept

```

```

user@P2#
set policy-options policy-statement prefix-sid term 1 from route-filter 192.168.255.2/32
exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128
index 1282
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128
node-segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129
index 1292
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129
node-segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1002
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

set policy-options policy-statement prefix-sid term 2 from route-filter
2001:db8:192:168:255::2/128 exact
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128
index 4282
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128
node-segment

```

```

set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129
index 4292
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129
node-segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment index 4002
set policy-options policy-statement prefix-sid term 2 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 2 then accept

```

```

user@PE2#
set policy-options policy-statement prefix-sid term 1 from route-filter 192.168.255.3/32
exact
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128
index 1283
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 128
node-segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129
index 1293
set policy-options policy-statement prefix-sid term 1 then prefix-segment algorithm 129
node-segment
set policy-options policy-statement prefix-sid term 1 then prefix-segment index 1003
set policy-options policy-statement prefix-sid term 1 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 1 then accept

set policy-options policy-statement prefix-sid term 2 from route-filter
2001:db8:192:168:255::3/128 exact
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128
index 4283
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 128
node-segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129
index 4293
set policy-options policy-statement prefix-sid term 2 then prefix-segment algorithm 129
node-segment
set policy-options policy-statement prefix-sid term 2 then prefix-segment index 4003
set policy-options policy-statement prefix-sid term 2 then prefix-segment node-segment
set policy-options policy-statement prefix-sid term 2 then accept

```



17. On all routers define the *RED* and *BLUE* MPLS administration groups, and assign the desired color to each interface. You also enable ICMP tunneling to allow trace route support in the context of MPLS based Layer 3 VPNs.

```
user@PE1#
set protocols mpls admin-groups RED 0
set protocols mpls admin-groups BLUE 1
set protocols mpls icmp-tunneling set protocols mpls interface ge-0/0/0.0 admin-group RED
set protocols mpls interface ge-0/0/1.0 admin-group BLUE
```

```
user@P1#
set protocols mpls admin-groups RED 0
set protocols mpls admin-groups BLUE 1
set protocols mpls icmp-tunneling      set protocols mpls interface ge-0/0/0.0 admin-group RED
set protocols mpls interface ge-0/0/1.0 admin-group BLUE
set protocols mpls interface ge-0/0/2.0 admin-group BLUE
```

```
user@P2#
set protocols mpls admin-groups RED 0
set protocols mpls admin-groups BLUE 1
set protocols mpls icmp-tunneling      set protocols mpls interface ge-0/0/0.0 admin-group BLUE
set protocols mpls interface ge-0/0/1.0 admin-group BLUE
set protocols mpls interface ge-0/0/2.0 admin-group BLUE
```

```
user@PE2#
set protocols mpls admin-groups RED 0
set protocols mpls admin-groups BLUE 1
set protocols mpls icmp-tunneling      set protocols mpls interface ge-0/0/0.0 admin-group BLUE
set protocols mpls interface ge-0/0/1.0 admin-group BLUE
```

18. Configure the FADs at the ingress PE device (PE1) under the *routing-options* hierarchy. In this case you assign flex algorithm 128 to optimize the path based on the *delay-metric* and 129 to optimize on the *igp-metric*. In this example, flex algorithm 128 must take only blue color paths, while flex algorithm 129 can take either a blue or a red color path. In this example you define the FADs at PE1 only as we focus only on the forwarding path from PE1 to PE2.

To support bidirectional flex path forwarding you will need to define the desired FADs on the PE2 device. The P routers don't require a FAD definition as the FAD is only used by the ingress node when calculating a path to the egress node.

```

user@PE1#
set routing-options flex-algorithm 128 definition metric-type delay-metric
set routing-options flex-algorithm 128 definition spf
set routing-options flex-algorithm 128 definition admin-group include-any BLUE

set routing-options flex-algorithm 129 definition metric-type igp-metric
set routing-options flex-algorithm 129 definition spf
set routing-options flex-algorithm 129 definition admin-group include-any RED
set routing-options flex-algorithm 129 definition admin-group include-any BLUE

```

**19.** Enter `commit` to from the configuration mode.

### ***Results***

Check the results of the configuration:

```
user@PE1# show interfaces
```

```

ge-0/0/0 {
  description To_R1;
  unit 0 {
    family inet {
      address 10.0.1.10/24;
    }
    family iso;
    family inet6 {
      address 2001:db8:10:0:1::10/80;
    }
    family mpls {
      maximum-labels 16;
    }
  }
}
ge-0/0/1 {
  description To_R2;
  unit 0 {
    family inet {

```

```

        address 10.0.2.10/24;
    }
    family iso;
    family inet6 {
        address 2001:db8:10:0:2::10/80;
    }
    family mpls {
        maximum-labels 16;
    }
}
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.255.10/32;
            address 127.0.0.1/32;
        }
        family iso {
            address 49.0001.000a.0a0a.0a00;
        }
        family inet6 {
            address 2001:db8:192:168:255::10/128;
        }
    }
    unit 1 {
        family inet {
            address 172.16.10.1/32;
        }
        family inet6 {
            address 2001:db8:172:16:10::1/128;
        }
    }
    unit 2 {
        family inet {
            address 172.16.10.2/32;
        }
        family inet6 {
            address 2001:db8:172:16:10::2/128;
        }
    }
}
}

```

```
user@PE1# show policy-options
```

```

policy-statement pplb {
  then {
    load-balance per-packet;
  }
}
policy-statement prefix-sid {
  term 1 {
    from {
      route-filter 192.168.255.10/32 exact;
    }
    then {
      prefix-segment {
        algorithm 128 index 1280 node-segment;
        algorithm 129 index 1290 node-segment;
        index 1000;
        node-segment;
      }
      accept;
    }
  }
  term 2 {
    from {
      route-filter 2001:db8:192:168:255::10/128 exact;
    }
    then {
      prefix-segment {
        algorithm 128 index 4280 node-segment;
        algorithm 129 index 4290 node-segment;
        index 4000;
        node-segment;
      }
      accept;
    }
  }
}
policy-statement v6vpn1_res_map1 {
  from {
    route-filter 2001:db8:172:16:1::/80 orlonger;
  }
  then {

```

```

        accept;
        resolution-map map1;
    }
}
policy-statement v6vpn2_res_map1 {
    from {
        route-filter 2001:db8:172:16:2::/80 orlonger;
    }
    then {
        accept;
        resolution-map map1;
    }
}
policy-statement vpn1_res_map1 {
    term 1 {
        from {
            route-filter 172.16.1.0/24 orlonger;
        }
        then {
            accept;
            resolution-map map1;
        }
    }
}
policy-statement vpn2_res_map1 {
    term 1 {
        from {
            route-filter 172.16.2.0/24 orlonger;
        }
        then {
            accept;
            resolution-map map1;
        }
    }
}
resolution-map map1 {
    mode ip-color;
}

```

user@PE1# show protocols

```

bgp {
  group to-RRv6 {
    type internal;
    local-address 2001:db8:192:168:255::10;
    import [ v6vpn1_res_map1 v6vpn2_res_map1 ];
    family inet6 {
      unicast {
        extended-nexthop-color;
      }
    }
    family inet6-vpn {
      unicast;
    }
    neighbor 2001:db8:192:168:255::2;
  }
  group to-RR {
    type internal;
    local-address 192.168.255.10;
    import [ vpn1_res_map1 vpn2_res_map1 ];
    family inet {
      unicast {
        extended-nexthop-color;
      }
    }
    family inet-vpn {
      unicast;
    }
    family traffic-engineering {
      unicast;
    }
    neighbor 192.168.255.2;
  }
}
isis {
  interface ge-0/0/0.0 {
    level 2 {
      post-convergence-lfa {
        node-protection;
      }
    }
  }
}

```

```

        delay-measurement {
            advertisement {
                periodic {
                    threshold 100;
                }
            }
        }
        point-to-point;
    }
interface ge-0/0/1.0 {
    level 2 {
        post-convergence-lfa {
            node-protection;
        }
    }
    delay-measurement {
        advertisement {
            periodic {
                threshold 100;
            }
        }
    }
    point-to-point;
}
interface lo0.0 {
    passive;
}
source-packet-routing {
    srgb start-label 80000 index-range 5000;
    flex-algorithm [ 128 129 ];
}
level 1 disable;
backup-spf-options {
    use-post-convergence-lfa maximum-backup-paths 8;
    use-source-packet-routing;
}
traffic-engineering {
    l3-unicast-topology;
    advertisement always;
}
export prefix-sid;
}
mpls {

```

```

traffic-engineering;
admin-groups {
    RED 0;
    BLUE 1;
}
icmp-tunneling;
interface all;
interface fxp0.0 {
    disable;
}
interface ge-0/0/0.0 {
    admin-group RED;
}
interface ge-0/0/1.0 {
    admin-group BLUE;
}
}

```

user@PE1# show routing-options

```

flex-algorithm 128 {
    definition {
        metric-type delay-metric;
        spf;
        admin-group include-any BLUE;
    }
}
flex-algorithm 129 {
    definition {
        metric-type igp-metric;
        spf;
        admin-group include-any [ RED BLUE ];
    }
}
router-id 192.168.255.10;
autonomous-system 64512;
forwarding-table {
    export pplb;
    ecmp-fast-reroute;
    chained-composite-next-hop {
        ingress {

```



```

        l3vpn;
    }
}

```

user@PE1# show routing-instances

```

vpn1 {
    instance-type vrf;
    interface lo0.1;
    route-distinguisher 64512:1;
    vrf-target target:64512:1;
    vrf-table-label;
}
vpn2 {
    instance-type vrf;
    interface lo0.2;
    route-distinguisher 64512:2;
    vrf-target target:64512:2;
    vrf-table-label;
}

```

user@PE1# show services rpm

```

twamp {
    server {
        authentication-mode none;
        light;
    }
}

```

## Verification

### IN THIS SECTION

- [Verify IS-IS Adjacencies | 644](#)
- [Verify IS-IS Database | 645](#)
- [Verify BGP Peering | 646](#)

- [Verify Color Community on VPN Routes | 648](#)
- [Verify junos-rti-tc-<color>.inet.0 Routing Table | 649](#)
- [Verify TWAMP Operation | 651](#)
- [Verify Route Resolution | 653](#)
- [Verify Forwarding Paths | 655](#)

**Verify IS-IS Adjacencies**

**IN THIS SECTION**

- [Purpose | 644](#)
- [Action | 644](#)
- [Meaning | 644](#)

**Purpose**

Verify expected IS-IS adjacencies on the routing devices.

**Action**

From operational mode, enter the `show isis adjacency` command.

```
user@PE1> show isis adjacency
```

Interface	System	L State	Hold (secs)	SNPA
ge-0/0/0.0	P1	2 Up	26	
ge-0/0/1.0	P2	2 Up	25	

**Meaning**

The output indicates that PE1 has successfully formed IS-IS adjacencies on its `ge-0/0/0.0` and `ge-0/0/1.0` interfaces, which attach to their P1 and P2 routers, respectively.

## Verify IS-IS Database

### IN THIS SECTION

- Purpose | 645
- Action | 645
- Meaning | 646

### **Purpose**

Verify that link delay parameters are present in the IS-IS database.

### **Action**

Use the `show isis database extensive | match delay` operational command.

```
user@PE1> show isis database extensive | match delay
```

```
Unidirectional link delay: 1041
  Min unidirectional link delay: 841
  Max unidirectional link delay: 1885
  Unidirectional delay variation: 71
  Unidirectional link delay: 2469
  Min unidirectional link delay: 766
  Max unidirectional link delay: 15458
  Unidirectional delay variation: 129
Unidirectional link delay: 20000
Min unidirectional link delay: 20000
Max unidirectional link delay: 20000
Unidirectional delay variation: 20000
  Unidirectional link delay: 1272
  Min unidirectional link delay: 628
  Max unidirectional link delay: 3591
  Unidirectional delay variation: 1559
  Unidirectional link delay: 8470
  Min unidirectional link delay: 855
  Max unidirectional link delay: 52934
  Unidirectional delay variation: 7900
```

```

Unidirectional link delay: 5736
Min unidirectional link delay: 3650
Max unidirectional link delay: 7946
Unidirectional delay variation: 4416
Unidirectional link delay: 2312
Min unidirectional link delay: 740
Max unidirectional link delay: 14227
Unidirectional delay variation: 3144
Unidirectional link delay: 1233
Min unidirectional link delay: 711
Max unidirectional link delay: 2833
Unidirectional delay variation: 366
Unidirectional link delay: 928
Min unidirectional link delay: 844
Max unidirectional link delay: 1042
Unidirectional delay variation: 143
Unidirectional link delay: 7570
Min unidirectional link delay: 761
Max unidirectional link delay: 61926
Unidirectional delay variation: 27290

```

### ***Meaning***

The output displays the dynamic delay that is associated with the various interfaces in the topology. The highlighted portion of the output specifies the static delay of 20000 microseconds that is configured on the P2 to PE2 link. The statically configured delay value is significantly higher than any of the dynamic delay measurements. This large delay is configured to make it easy to predict the delay optimized blue path through the network.

### ***Verify BGP Peering***

#### **IN THIS SECTION**

- Purpose | 646
- Action | 647
- Meaning | 647

### ***Purpose***

Verify that both PEs have successfully established IPv4 and IPv6 peering sessions to the route reflector.

## Action

Use the `show bgp summary operational` command. In this case we run the command on P2, the route reflector, as it provides a convenient location to confirm both peering sessions from both PEs using a single command.

```
user@P2 show bgp summary
```

```
Threading mode: BGP I/O
Default eBGP mode: advertise - accept, receive - accept
Groups: 2 Peers: 4 Down peers: 0
```

Table	Tot Paths	Act Paths	Suppressed	History	Damp	State	Pending
inet6.0	0	0	0	0	0	0	0
bgp.l3vpn-inet6.0	6	6	0	0	0	0	0
inet.0	0	0	0	0	0	0	0
bgp.l3vpn.0	6	6	0	0	0	0	0

```

Peer          AS      InPkt    OutPkt    OutQ    Flaps  Last Up/Dwn  State|#Active/
Received/Accepted/Damped...
192.168.255.3  64512    2511     2489      0       0    18:49:42  Establ
  inet.0: 0/0/0/0
  bgp.l3vpn.0: 4/4/4/0
192.168.255.10 64512    2511     2491      0       0    18:49:46  Establ
  inet.0: 0/0/0/0
  bgp.l3vpn.0: 2/2/2/0
2001:db8:192:168:255::3 64512    2512     2490      0       0    18:49:46  Establ
  inet6.0: 0/0/0/0
  bgp.l3vpn-inet6.0: 4/4/4/0
2001:db8:192:168:255::10 64512    2510     2490      0       0    18:49:42  Establ
  inet6.0: 0/0/0/0
  bgp.l3vpn-inet6.0: 2/2/2/0

```

## Meaning

The output confirms that all BGP peering sessions are established correctly. The display also confirms that Layer 3 VPN routes are being advertised/learned over these peering sessions.

## Verify Color Community on VPN Routes

### IN THIS SECTION

- Purpose | 648
- Action | 648
- Meaning | 649

### **Purpose**

Verify the VPN routes advertised by PE2 are correctly tagged with a color community.

### **Action**

Use the `show route detail <prefix> table <table-name> operational` command at PE1 to display details about a Layer 3 VPN route learned from PE2.

```
user@PE1 show route detail 172.16.1.0 table vpn1
```

```
vpn1.inet.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
172.16.1.0/24 (1 entry, 1 announced)
    *BGP      Preference: 170/-101
              Route Distinguisher: 64512:1
              Next hop type: Indirect, Next hop index: 0
              Address: 0xc5b9d5c
              Next-hop reference count: 3
              Source: 192.168.255.2
              Next hop type: Router, Next hop index: 0
              Next hop: 10.0.2.2 via ge-0/0/1.0 weight 0x1, selected
              Label operation: Push 81282
              Label TTL action: prop-ttl
              Load balance label: Label 81282: None;
              Label element ptr: 0xcbf1440
              Label parent element ptr: 0x0
              Label element references: 2
              Label element child references: 0
              Label element lsp id: 0
              Session Id: 0x0
```

```

Protocol next hop: 192.168.255.3-128<c>
Label operation: Push 16
Label TTL action: prop-ttl
Load balance label: Label 16: None;
Composite next hop: 0xbd50440 665 INH Session ID: 0x0
Indirect next hop: 0xc74e684 1048588 INH Session ID: 0x0
State: <Secondary Active Int Ext ProtectionCand>
Local AS: 64512 Peer AS: 64512
Age: 19:10:35 Metric2: 2204
Validation State: unverified
ORR Generation-ID: 0
Task: BGP_64512.192.168.255.2
Announcement bits (1): 0-KRT
AS path: I (Originator)
Cluster list: 192.168.255.2
Originator ID: 192.168.255.3
Communities: target:64512:1 color:0:128
Import Accepted
VPN Label: 16
Localpref: 100
Router ID: 192.168.255.2
Primary Routing Table: bgp.l3vpn.0
Thread: junos-main

```

### Meaning

The output confirms that a VPN prefix in the VPN1 routing instance has a color community `color:0:128` attached. In addition, you can confirm that the protocol next hop for this route is the loopback address of the PE2 router with an extended next hop that indexes a matching entry in the color table.

Though not shown, you can repeat this command for a prefix in the VPN2 table. You expect to find these routes have the `color:0:129` attached.

### Verify `junos-rti-tc-<color>.inet.0 Routing Table`

#### IN THIS SECTION

- Purpose | 650
- Action | 650
- Meaning | 651

## Purpose

Verify the `junos-rti-tc-<color>.inet.0` routing table is correctly populated with all router IDs (loopback addresses) showing support for both the 128 and 129 flex algorithms.



**NOTE:** IPv6 routes are supported via the `junos-rti-tc-<color>.inet6.0` table. You can verify this table using the same approach as shown in this section for the IPv4 color table.

## Action

Use the `show route table junos-rti-tc-<color>.inet.0` operational command.

```
user@PE1> show route table junos-rti-tc-<color>.inet.0
```

```
junos-rti-tc-<color>.inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
192.168.255.1-128<c>/64
```

```
    *[L-ISIS/14] 6d 14:40:37, metric 1527
```

```
    > to 10.0.2.2 via ge-0/0/1.0, Push 81281
```

```
192.168.255.1-129<c>/64
```

```
    *[L-ISIS/14] 6d 14:40:35, metric 10
```

```
    > to 10.0.1.1 via ge-0/0/0.0
```

```
        to 10.0.2.2 via ge-0/0/1.0, Push 81291
```

```
192.168.255.2-128<c>/64
```

```
    *[L-ISIS/14] 6d 14:40:40, metric 761
```

```
    > to 10.0.2.2 via ge-0/0/1.0
```

```
192.168.255.2-129<c>/64
```

```
    *[L-ISIS/14] 6d 14:40:35, metric 10
```

```
    > to 10.0.2.2 via ge-0/0/1.0
```

```
        to 10.0.1.1 via ge-0/0/0.0, Push 81292
```

```
192.168.255.3-128<c>/64
```

```
    *[L-ISIS/14] 6d 14:40:37, metric 2382
```

```
    > to 10.0.2.2 via ge-0/0/1.0, Push 81283
```

```
192.168.255.3-129<c>/64
```

```
    *[L-ISIS/14] 6d 14:40:35, metric 20
```

```
    > to 10.0.1.1 via ge-0/0/0.0, Push 81293
```



```
to 10.0.2.2 via ge-0/0/1.0, Push 81293
```

**Meaning**

The output displays the routes in the junos-rti-tc-<color>.inet.0 route table. The highlighted portion indicates the two routes originate from PE2. The 192.168.255.3-128<c> route has only one possible path and takes the ge-0/0/1.0 interface to P2 as a next hop. Recall that the 128 flex algorithm must use blue links, and from the perspective of PE1, which leaves only the blue colored ge-0/0/1 interface as a viable path.

In contrast, the route for 192.168.255.3-129<c> is able to load balance over both the ge-0/0/0.0 interfaces to P1 and the ge-0/0/1.0 to P2. Recall that this path for flex algorithm can take any path that is either blue or red, thus can use either of its interfaces when forwarding to its associated destination.

**Verify TWAMP Operation**

IN THIS SECTION

Purpose | 651

Action | 651

Meaning | 652

**Purpose**

Verify that TWAMP probes are operating between routers with dynamic link delay configured.

**Action**

Use the show services rpm twamp client operational mode command.

```
user@PE1> show services rpm twamp client
```

Connection Name	Session Name	Sender address	Sender port	Reflector address	Reflector port
-----------------	--------------	----------------	-------------	-------------------	----------------

__r__8	__r__9	10.0.1.10	56570 <b>10.0.1.1</b>	862
__r__10	__r__11	10.0.2.10	64074 <b>10.0.2.2</b>	862

### **Meaning**

The highlighted portion of the output indicates that PE1 has two TWAMP neighbors: P2 (10.0.1.2) and P1 (10.0.1.1).

If desired use the `show services rpm twamp client probe-results operational mode` command to see the current and historical delay measurement values.

```
user@PE1> show services rpm twamp client probe-results
```

```
root@PE1# run show services rpm twamp client probe-results
Owner: __r__12, Test: __r__13
TWAMP-Server-Status: Light, Number-Of-Retries-With-TWAMP-Server: 0
Reflector address: 10.0.2.2, Reflector port: 862, Sender address: 10.0.2.10, sender-port:
57270
Test size: 10 probes
Probe results:
  Response received
  Probe sent time: Thu May  6 14:43:26 2021
  Probe rcvd/timeout time: Thu May  6 14:43:26 2021
  Rtt: 1931 usec, Egress jitter: 259 usec, Ingress jitter: 96 usec, Round trip jitter: 353
usec
  Egress interarrival jitter: 5489 usec, Ingress interarrival jitter: 855 usec, Round trip
interarrival jitter: 6076 usec
Results over current test:
  Probes sent: 8, Probes received: 8, Loss percentage: 0.000000
  Measurement: Round trip time
    Samples: 8, Minimum: 1576 usec, Maximum: 13289 usec, Average: 6100 usec, Peak to peak:
11713 usec, Stddev: 4328 usec,
    Sum: 48797 usec
  Measurement: Ingress delay
    Samples: 2, Minimum: 8466 usec, Maximum: 8488 usec, Average: 8477 usec, Peak to peak: 22
usec, Stddev: 11 usec,
    Sum: 16954 usec
  Measurement: Egress delay
    Samples: 2, Minimum: 118 usec, Maximum: 4801 usec, Average: 2460 usec, Peak to peak:
4683 usec, Stddev: 2342 usec,
```

```

    Sum: 4919 usec
    Measurement: Positive egress jitter
    Samples: 4, Minimum: 259 usec, Maximum: 11250 usec, Average: 4465 usec, Peak to peak:
10991 usec, Stddev: 4225 usec,
    Sum: 17859 usec
    Measurement: Negative egress jitter
    Samples: 4, Minimum: 201 usec, Maximum: 6564 usec, Average: 4467 usec, Peak to peak:
6363 usec, Stddev: 2566 usec,
    Sum: 17869 usec
    Measurement: Positive ingress jitter
    Samples: 5, Minimum: 96 usec, Maximum: 4954 usec, Average: 1431 usec, Peak to peak: 4858
usec, Stddev: 1843 usec,
    Sum: 7155 usec
    Measurement: Negative ingress jitter
    Samples: 3, Minimum: 202 usec, Maximum: 4990 usec, Average: 2340 usec, Peak to peak:
4788 usec, Stddev: 1988 usec,
    Sum: 7021 usec
    Measurement: Positive round trip jitter
    Samples: 4, Minimum: 353 usec, Maximum: 11585 usec, Average: 5827 usec, Peak to peak:
11232 usec, Stddev: 4797 usec,
    Sum: 23309 usec
    Measurement: Negative round trip jitter
    Samples: 4, Minimum: 2056 usec, Maximum: 9734 usec, Average: 5831 usec, Peak to peak:
7678 usec, Stddev: 2776 usec,
    Sum: 23325 usec
    Results over last test:
. . .

```

### ***Verify Route Resolution***

#### **IN THIS SECTION**

- Purpose | 653
- Action | 654
- Meaning | 655

#### ***Purpose***

Verify the routes for the VPN1 and VPN2 resolve over the expected flex algorithm paths.

**Action**

Use the `show route operational mode` command.

```
user@PE1> show route 172.16.1.0
```

```
inet.0: 18 destinations, 18 routes (17 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
. . .
vpn1.inet.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.1.0/24      *[BGP/170] 6d 16:32:32, localpref 100, from 192.168.255.2
                  AS path: I, validation-state: unverified
                  > to 10.0.2.2 via ge-0/0/1.0, Push 16, Push 81287(top)
```

```
user@PE1> show route 172.16.2.0
```

```
inet.0: 18 destinations, 18 routes (17 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both. . .

vpn2.inet.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.2.0/24      *[BGP/170] 6d 16:36:02, localpref 100, from 192.168.255.2
                  AS path: I, validation-state: unverified
                  to 10.0.1.1 via ge-0/0/0.0, Push 17, Push 81297(top)
                  > to 10.0.2.2 via ge-0/0/1.0, Push 17, Push 81297(top)
```

### Meaning

The highlighted output indicates that on the PE1 device, the 172.16.1.0 route for VPN1 uses FAD 128 taking only the blue color path, which makes P1 (10.0.2.2) its next hop while the route for VPN2, 172.16.2.0 uses FAD 129, which means it can take the red color path either through ge-0/0/0.0 interface to P1>PE2 or through the ge-0/0/1.0 interface to P2> PE2. This is also true for IPv6 routes, as shown here for VPN1:

```
user@PE1> show route 2001:db8:172:16:1::/80
```

```
vpn1.inet6.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
```

```
+ = Active Route, - = Last Active, * = Both
```

```
2001:db8:172:16:1::/80
```

```
*[BGP/170] 01:26:27, localpref 100, from 2001:db8:192:168:255::2
```

```
AS path: I, validation-state: unverified
```

```
> to fe80::5668:a5ff:fed1:21d9 via ge-0/0/1.0, Push 16, Push 84287(top)
```

The IPv6 route from VPN1 resolves to the same forwarding path as its IPv4 counterpart, which makes sense as they are both using flex algorithm 128 to force the use of blue links with delay optimization. Recall that you configured PE2, the source of these routes, to use a label base of 1287 for IPv4 routes and 4287 for IPv6 routes, and that the source-packet-routing srgb start-label to 8000. As a result the IPv4 route from VPN1 has a label of 81287 while the IPv6 route from VPN1 uses 84287.

### Verify Forwarding Paths

#### IN THIS SECTION

● Purpose | 655

● Action | 656

● Meaning | 657

### Purpose

Verify the routes for VPN1 and VPN2 are forwarded over the expected flex algorithm paths.

## Action

Use the ping and trace route operational mode commands to verify reachability, and to confirm the IPv4 forwarding path used by PE1 when sending traffic to VPN destinations as PE2.



**NOTE:** The use of static routes with a receive next hop at PE2 allows you to ping the remote routes. You can expect the last hop of the trace route to timeout, however, as trace route processing is not supported when targeting an IPv4 static receive route.

```
user@PE1> ping 172.16.1.0 routing-instance vpn1 count 2
```

```
PING 172.16.1.0 (172.16.1.0): 56 data bytes
64 bytes from 172.16.1.0: icmp_seq=0 ttl=63 time=6.617 ms
64 bytes from 172.16.1.0: icmp_seq=1 ttl=63 time=33.849 ms

--- 172.16.1.0 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 6.617/20.233/33.849/13.616 ms
```

```
user@PE1> traceroute 172.16.1.0 routing-instance vpn1 no-resolve
```

```
traceroute to 172.16.1.0 (172.16.1.0), 30 hops max, 52 byte packets
 1  10.0.2.2 (10.0.2.2)  4.729 ms  4.698 ms  4.559 ms
    MPLS Label=81282 CoS=0 TTL=1 S=0
    MPLS Label=16 CoS=0 TTL=1 S=1
 2  10.0.12.1 (10.0.12.1)  8.524 ms  7.780 ms  4.338 ms
    MPLS Label=81282 CoS=0 TTL=1 S=0
    MPLS Label=16 CoS=0 TTL=2 S=1
 3  * * *
```

```
*^C
user@PE1>
```

```
user@PE1> ping 172.16.2.0 routing-instance vpn1 count 2
```

```
PING 172.16.2.0 (172.16.2.0): 56 data bytes
64 bytes from 172.16.2.0: icmp_seq=0 ttl=63 time=31.723 ms
64 bytes from 172.16.2.0: icmp_seq=1 ttl=63 time=3.873 ms

--- 172.16.2.0 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 3.873/17.798/31.723/13.925 ms
```

```
user@PE1> traceroute 172.16.2.0 routing-instance vpn2 no-resolve
```

```
traceroute to 172.16.2.0 (172.16.2.0), 30 hops max, 52 byte packets
 1  10.0.1.1  7.102 ms  8.746 ms  7.820 ms
    MPLS Label=81292 CoS=0 TTL=1 S=0
    MPLS Label=17 CoS=0 TTL=1 S=1
 2  * * *
*^C
user@PE1>
```

### ***Meaning***

The output indicates that the expected forwarding paths are used. For example, the trace route for the 172.16.1.0/24 route in VPN1 shows that blue paths are used, and that the high-delay link between P2 and PE2 is avoided. This confirms that flex algorithm prefers a path with an extra hop if it results in a reduction of end-to-end path latency. In this case the 10.0.12.0 link between P2 and P1 is used while the direct link between P2 and PE2 is avoided.

In contrast, the path taken for the 172.16.2.0/24 route, associated with VPN2 and flex algorithm 129, is able to take either of the direct paths between PE1 and PE2. In this case the forwarding path is from PE1 to P1 and then to the destination (PE2), where as noted the last hop times out. This timeout on the last hop does not occur for routes that point to a CE device (as opposed to the static receive routes used in this example).

Though not shown here for brevity, you expect the same forwarding paths for trace routes to the IPv6 VPN routes based on whether they are mapped to flex algorithm 128 or 129, which in this example means associated with VPN1 versus VPN2, respectively.

## Configuring OSPF Link Delay and Delay Normalization on an OSPF Interface

In IP networks, the bulk of traffic often goes through the core network, which reduces costs but might result in increased latency. Business traffic, however, often benefits from the ability to make path-selection decisions based on other performance metrics, such as path latency, rather than relying on the traditional path optimization based simply on IGP metrics. Optimizing a path to reduce latency can greatly benefit applications like real-time voice and video. It can also enable high performance access to financial market data where milliseconds can translate into significant gains or losses.

You can enable OSPF link delay in IP networks. You can achieve minimum IGP metric paths by configuring OSPF with the appropriate link cost using the default OSPF algorithm. Doing so optimizes paths to the endpoint that are based strictly on the sum of the link metrics. By using the OSPF delay flex algorithm you can optimize paths based on their end-to-end delay.

Link delay can be dynamically measured using Two-Way Active Measurement Probes (TWAMP). The routers then flood their link delay parameters. The routers in the area store these parameters in the shared Link State Database (LSDB). Ingress nodes run an SPF algorithm against the LSDB to compute paths that are optimized on various attributes, such as link colors, IGP metric, traffic-engineering (TE) metric.

To configure link delay measurement for an OSPF Interface:

1. Create an OSPF area.

```
[edit protocols]
user@host#set protocols ospf area area-id
```

For example:

```
[edit protocols]
user@host#set protocols ospf area 0.0.0.0
```

2. Specify the interface.

```
[edit protocols ospf area 0.0.0.0]
user@host#set interface interface-name
```



For example:

```
[edit protocols ospf area 0.0.0.0]
user@host#set interface ge-0/0/0.0
```

3. Configure dynamic OSPF link delay-measurement on the OSPF interface of the device.

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement
```

4. Configure the delay-measurement advertisement on the OSPF interface of the device. You can either configure accelerated or periodic advertisement.

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement advertisement (accelerated | periodic)
```



**NOTE:** Accelerated advertisement is disabled by default. To configure accelerated advertisement, configure the threshold percentage.

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement advertisement accelerated threshold percentage
```

5. To configure periodic advertisement, you can either configure interval or the threshold percentage.

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement advertisement periodic (interval interval seconds |
threshold threshold percentage)
```

For example: To configure periodic interval:

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement advertisement periodic interval 35
```

For example: To configure periodic threshold:

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement advertisement periodic threshold 100)
```

6. (Optional) Specify the probe count.

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement probe-count seconds
```

For example:

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement probe-count 10
```

7. (Optional) Specify the probe interval.

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement probe-interval seconds
```

For example:

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement probe-interval 100
```

8. (Optional) Specify the normalize interval and offset values.

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement normalize interval microseconds offset microseconds seconds
```

For example:

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-measurement normalize interval 50 offset 10
```

9. Enter `commit` from the configuration mode.

To configure delay metric for an OSPF Interface:

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
user@host#set delay-metric microseconds
```

For example:

```
[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]  
user@host#set delay-metric 20000
```

Enter `commit` from the configuration mode.

To verify your configuration results, use the `show protocols operational` command.

```
user@host# show protocols
```

```
ospf {  
    area 0.0.0.0  
        interface ge-0/0/0.0 {  
            delay-measurement {  
advertisement {  
    accelerated {  
        threshold 100;  
    }  
    periodic {  
        interval 35;  
        threshold 100;  
    }  
}  
    probe-count 10;  
    probe-interval 100;  
}  
        delay-metric {  
            20000;  
        }  
}  
    normalize interval 50 offset 10;
```

To verify that link delay parameters are present in the OSPF database use the `show ospf database extensive | match delay operational` command.

```
user@host> show ospf database extensive | match delay
```

```
Unidirectional link delay: 20000
Min unidirectional link delay: 20000
Max unidirectional link delay: 20000
Unidirectional delay variation: 20000
```

The output displays the delay of 20000 microseconds that is configured on the interface.

To verify the normalized delay value, use the `show ospf interface extensive operational` command.

```
user@host> show ospf interface extensive
```

```
root@R0# run show ospf interface ge-0/0/0.0 extensive
Interface          State   Area        DR ID        BDR ID        Nbrs
ge-0/0/0.0          PtToPt 0.0.0.0      0.0.0.0      0.0.0.0        1
  Type: P2P, Address: 21.0.1.1, Mask: 255.255.255.0, MTU: 1500, Cost: 1
  Adj count: 1
  Hello: 10, Dead: 40, ReXmit: 5, Not Stub
  Auth type: None
  Protection type: None
  Topology default (ID 0) -> Cost: 1
  Unidirectional link delay: 1616
  Min unidirectional link delay: 712
  Max unidirectional link delay: 8125
  Unidirectional delay variation: 2992
  Delay Normalization: Enabled (Interval: 50, Offset: 10)
  Normalized Delay: 860 (Last Normalized: Mon Jul 24 10:03:41)
```

The output displays that the unidirectional link delay 1616 microseconds has been normalized into a value (860) that OSPF uses in SPF computations for flexible algorithm. Last Normalized: Mon Jul 24 10:03:41 indicates the timestamp of the most recent normalization calculation.

SEE ALSO

<a href="#"><i>delay-measurement (Protocols OSPF)</i></a>
<a href="#"><i>delay-metric (Protocols OSPF)</i></a>
<a href="#">Understand Two-Way Active Measurement Protocol</a>

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
25.4R1	Starting in Junos OS and Junos OS Evolved Release 25.4R1, you can use delay normalization for OSPFv2 to compute and advertise a normalized delay metric for flexible algorithm, to improve path-selection consistency across all IGP instances.
21.4R1	Starting in Junos OS Release 21.4R1, you can get the measurement of various performance metrics in IP networks, by using probe messages.

How to Enable Strict SPF SIDs and IGP Shortcut

IN THIS SECTION

- [Understanding Strict SPF \(SR-Algo 1\) and IGP Shortcuts | 663](#)
- [Example: Configure Strict SPF SIDs and Enable IGP Shortcuts in SPRING for IS-IS Protocol | 666](#)

Understanding Strict SPF (SR-Algo 1) and IGP Shortcuts

<p>IN THIS SECTION</p> <ul style="list-style-type: none"><li>● <a href="#">Benefits of Strict SPF (SR-Algo 1) and IGP Shortcuts   664</a></li><li>● <a href="#">Overview of Strict SPF (SR-Algo 1) and IGP Shortcuts   664</a></li><li>● <a href="#">What's Next?   666</a></li></ul>
---

Strict SPF (SR-Algo 1) and IGP shortcut provides the following benefits

### Benefits of Strict SPF (SR-Algo 1) and IGP Shortcuts

- Enhances segment routing capabilities.
- Helps to avoid loops by creating SR-TE tunnel to forward the traffic using the shortest IGP path.
- Ability to use SR-Algo 1 (strict SPF) along with SR-Algo 0 (default SPF) by default, when you enable SPRING.

### Overview of Strict SPF (SR-Algo 1) and IGP Shortcuts

Segment routing (SR) simplifies operations and reduces resource requirements in the network by removing network state information from intermediate routers and placing path information into packet headers at the ingress node. However, in some cases, when there are nested SR-TE tunnels present and devices forward traffic over these SR-TE tunnel, traffic might loop, cause congestion, and not forward traffic over the shortest IGP path.

You can advertise SR algorithm 1 (strict SPF) and use the strict SPF SIDs to create SR-TE tunnels. Such SR-TE tunnels use only the strict path SPF instead of the local policy to reach the tunnel endpoint. You can specify prefixes in the import policy, based on which the tunnels redirect the traffic to a certain destination. Additionally, you can use SR-Algo 1 (strict SPF) along with SR-Algo 0 (default SPF) by default when you enable SPRING.

You can advertise strict-SPF SIDs in IS-IS LSPDU and use these SIDs to create SR-TE tunnel to forward the traffic through the shortest IGP path while not causing loops. Labeled IS-IS routes will then use the tunnel with the pre-defined shortcut statement at the `inet-mpls` family or `inet6-mpls` family configuration when you prefer `spring-te` tunnel.

The following illustration depicts the difference between SR-TE tunnels created without strict SPF SIDs and SR-TE tunnels created by using strict SPF (SR-Algo 1) SIDs:

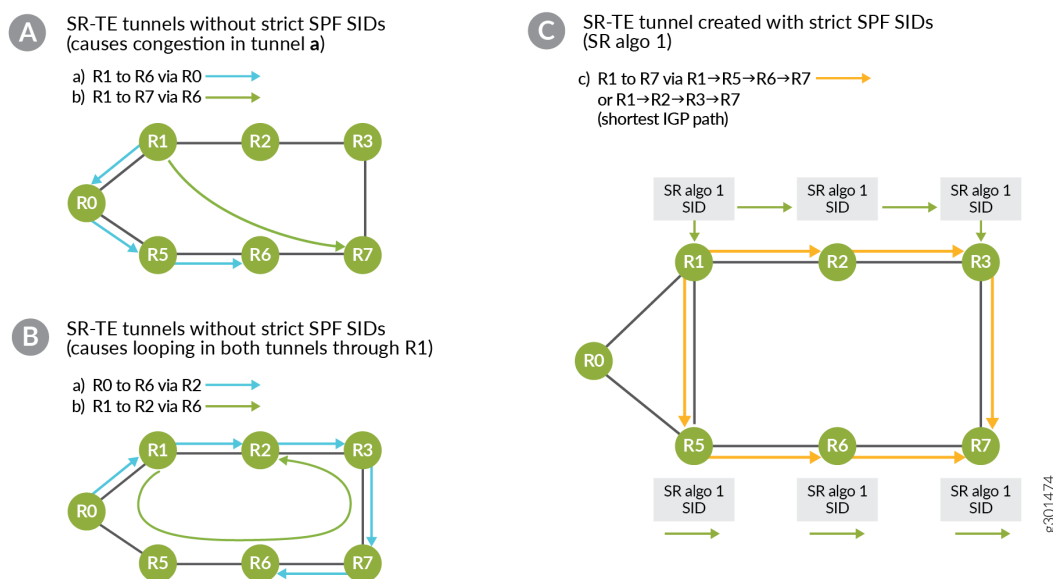


Figure A shows a network topology where SR-TE tunnel is not created using shortest IGP path to forward a traffic when a pre-existing SR-TE tunnel (or RSVP tunnel) is selected as ingress at R1. Two SR-TE tunnels exist in this topology. One from R1 to R6 (tunnel a, blue colored) via R0 and another tunnel is R1 to R7 (tunnel b, green colored) via R6. In this case, tunnel (b) is not created using the shortest IGP path. Thus, instead of taking the existing tunnel to reach R6 and then forwarding to R7, since, `inet-mpls shortcut` statement is enabled on R1, label IS-IS route uses the SR-TE tunnel (a) to forward the traffic destined to R7 avoiding the shortest IGP path, resulting traffic congestion on tunnel (a).

Figure B shows a topology where traffic loops. When the labeled IS-IS route chooses SR-TE tunnel as ingress and redirected to another SR-TE tunnel, then traffic will loop. In this topology we have two SR-TE tunnels, one from R0 to R6 via R2 and another tunnel is from R1 to R2 via R6. For a packet sent from R0 to R6 node, at R0 if this node picks SR-TE tunnel as ingress for the destination 2.2.2.6, it will push R2 label and forward to R1. At R1, another SR-TE tunnel is present via R6 with a label in `mpls.0` table. When R1 receives this traffic to reach R2, it will use L-ISIS route shortcut over SR-TE tunnel and push R6 with the same label then forward to R0 node. At R0, the top label is the same as R6, which means that if the SR-TE tunnel again then it will push R2 label and forward the traffic to R1, which will loop.

Figure C shows the SR-TE tunnels created using Strict SPF SIDs that now supports SR-Algo 1 along with the pre-existing SR-Algo 0. Strict-SPF SID routes are installed in IS-IS only if the next-hop node is also capable of SR algo 1. Else, the traffic will be dropped. If you created the SR-TE tunnel using strict SPF SIDs and if anywhere on the path where a device did not advertise support for SR Algo 1, the tunnel will stay down. When tunnel is created using Strict SPF SIDs it will take the shortest IGP path to reach another tunnel endpoint, and thereby, avoids congestion. In a scenario where traffic loops (as shown in figure 2), the strict-SPF SIDs will be advertised in IS-IS LSPDU only by each node that is participating in SR domain that supports SR Algo 1. There can be multiple SR-TE tunnels, either created by using Strict-SPF SIDs or normal SIDs. When the operator configures the statement “`use-for-shortcut`” before creating the explicit route object (ERO), tunnels are created using strict SPF SIDs.



**NOTE:** This topic provides only a configuration overview of SR-TE needed for IGP shortcuts.

### What's Next?

Stay tuned for more information about the working principles of SR-TE!

## Example: Configure Strict SPF SIDs and Enable IGP Shortcuts in SPRING for IS-IS Protocol

### IN THIS SECTION

- [Overview | 666](#)
- [Requirements | 666](#)
- [Configuration | 668](#)
- [Verification | 684](#)

### Overview

Typically, when there are nested SR-TE tunnels present in a network and devices forward traffic over these SR-TE tunnels, traffic might not get forwarded over the shortest IGP path. As a result, traffic might loop.

You can advertise SR algorithm and use the strict SPF SIDs to create SR-TE tunnels to forward the traffic using shortest IGP path to avoid loop. Labeled IS-IS route will now use this tunnel with the pre-defined shortcut knob present under `inet-mpls` family (or `inet6-mpls` family) when you enable `spring-te`.

### Requirements

### IN THIS SECTION

- [Topology | 667](#)

This example uses the following hardware and software components:



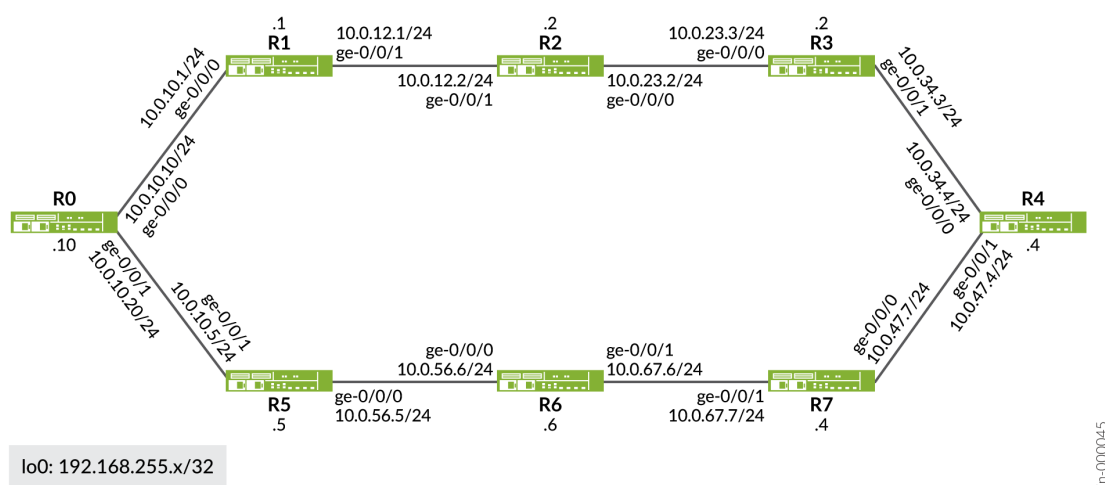
- Eight MX Series routers
- Junos OS Release 21.1R1 or later running on all devices

### Topology

In the following example, we are demonstrating how loops occur in a network with multiple SR-TE tunnels and how you can resolve it by using strict SPF SIDs created by SR Algorithm 1. The example topology has two SR-TE tunnels. Tunnel A from R0>R2>R6 and Tunnel B from R1>R6>R2.

On R0 a packet destined to R6 typically use the IGP shortest path: that is, R0>R5>R6. When you configure an SR-TE tunnel with its ingress node as R0 (tunnel A), the packet needs to go through R2 as its first hop (destination: R6 and label: 403002), which means the traffic destined to R6 needs to take the R0>R2>R6 path. To reach R2, the packet needs to reach R1 first on the R0—R1 interface with the first label 403002. The R2's label 403002 should get forwarded from R1>R2 with no changes to the label stack. However, there is a second SR-TE tunnel (tunnel B) configured on R1 (R1>R6>R2) with destination R2 and label 403006. The packet that came from R0 with top label as R2 (403002) on R1 ends up using the second tunnel to reach R6. But to reach R6 on R1, R1—R0 (R1>R0>R5>R6) is the interface it needs to use. Thus, the packet reaches R0 again and the whole process repeats, resulting in looping.

With the SR algorithm 1 activated on all devices, and its labels activated on the relevant devices, when the packet from the ingress device R0 to the destination device R6 reaches R1 (tunnel A), the packet gets forwarded to R2. Even though R1 has LSP configured to consider R6 as its next hop (tunnel B), it would instead take the IGP shortest path (R1>R2). From R2, it reaches R6 through Tunnel A.



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 668](#)
- [Enable Default SIDs \(Algorithm 0\) in SPRING | 677](#)
- [Enable Strict SPF SIDs \(Algorithm 1\) in SPRING | 679](#)
- [Results | 681](#)

### *CLI Quick Configuration*

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.



**NOTE:** Depending on the type of MPC in your MX Series routers you might need to explicitly enable enhanced IP services to support the IS-IS delay feature. When you commit the `set chassis network-services enhanced-ip` configuration statement, you will be prompted to reboot the system.

R0

```
set system host-name R0
set system ports console log-out-on-disconnect
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 10.0.10.10/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 8
set interfaces ge-0/0/1 unit 0 family inet address 10.0.10.20/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 8
set interfaces lo0 unit 0 family inet address 192.168.255.10/32
set interfaces lo0 unit 0 family iso address 49.1921.6825.5010.00
set interfaces lo0 unit 0 family mpls maximum-labels 8
set policy-options policy-statement sspf term 1 from route-filter 192.168.255.10/32 exact
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3100
set policy-options policy-statement sspf term 1 then prefix-segment index 3000
```

```

set policy-options policy-statement sspf term 1 then prefix-segment node-segment
set policy-options policy-statement sspf term 1 then accept
set protocols isis interface ge-0/0/0.0
set protocols isis interface ge-0/0/1.0
set protocols isis interface all
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 64000
set protocols isis source-packet-routing node-segment ipv4-index 1000
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis traffic-engineering tunnel-source-protocol spring-te
set protocols isis traffic-engineering family inet-mpls shortcuts
set protocols isis export sspf
set protocols mpls traceoptions file sspf-igp-short
set protocols mpls traceoptions file size 100m
set protocols mpls traceoptions file world-readable
set protocols mpls traceoptions flag ted-export
set protocols mpls traceoptions flag ted-import
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols rsvp interface all
deactivate protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
deactivate protocols rsvp interface fxp0.0 disable
set protocols source-packet-routing segment-list v4R0R7 h1 label 403102
set protocols source-packet-routing source-routing-path V4_R7 use-for-shortcut
set protocols source-packet-routing source-routing-path V4_R7 to 192.168.255.6
set protocols source-packet-routing source-routing-path V4_R7 primary v4R0R7
set routing-options router-id 192.168.255.10
set routing-options autonomous-system 65540

```

R1

```

set system host-name R1
set system ports console log-out-on-disconnect
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 10.0.10.1/24
set interfaces ge-0/0/0 unit 0 family iso

```

```

set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 8
set interfaces ge-0/0/1 unit 0 family inet address 10.0.12.1/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 8
set interfaces lo0 unit 0 family inet address 192.168.255.1/32
set interfaces lo0 unit 0 family iso address 49.1921.6825.5001.00
set interfaces lo0 unit 0 family mpls maximum-labels 8
set policy-options policy-statement sspf term 1 from route-filter 192.168.255.1/32 exact
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3101
set policy-options policy-statement sspf term 1 then prefix-segment index 3001
set policy-options policy-statement sspf term 1 then prefix-segment node-segment
set policy-options policy-statement sspf term 1 then accept
set protocols isis interface all
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 64000
set protocols isis source-packet-routing node-segment ipv4-index 1001
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis traffic-engineering tunnel-source-protocol spring-te
set protocols isis traffic-engineering family inet-mpls shortcuts
set protocols isis export sspf
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
set protocols source-packet-routing segment-list v4R1R2 h1 label 403106
set protocols source-packet-routing source-routing-path V4_R2 use-for-shortcut
set protocols source-packet-routing source-routing-path V4_R2 to 192.168.255.2
set protocols source-packet-routing source-routing-path V4_R2 primary v4R1R2
set routing-options router-id 192.168.255.1
set routing-options autonomous-system 65540

```

## R2

```

set system host-name R2
set system ports console log-out-on-disconnect
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 10.0.23.2/24

```

```

set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 8
set interfaces ge-0/0/1 unit 0 family inet address 10.0.12.2/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 8
set interfaces lo0 unit 0 family inet address 192.168.255.2/32
set interfaces lo0 unit 0 family iso address 49.1921.6825.5002.00
set interfaces lo0 unit 0 family mpls maximum-labels 8
set policy-options policy-statement sspf term 1 from route-filter 192.168.255.2/32 exact
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3102
set policy-options policy-statement sspf term 1 then prefix-segment index 3002
set policy-options policy-statement sspf term 1 then prefix-segment node-segment
set policy-options policy-statement sspf term 1 then accept
set protocols isis interface all
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 64000
set protocols isis source-packet-routing node-segment ipv4-index 1002
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis level 2 wide-metrics-only
set protocols isis level 1 wide-metrics-only
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis traffic-engineering tunnel-source-protocol spring-te
set protocols isis traffic-engineering family inet-mpls shortcuts
set protocols isis export sspf
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
set protocols source-packet-routing segment-list v4R2R6 h1 label 403100
set protocols source-packet-routing segment-list v4R2R6 h2 label 403107
set protocols source-packet-routing source-routing-path v4_R6 use-for-shortcut
set protocols source-packet-routing source-routing-path v4_R6 to 192.168.255.2
set protocols source-packet-routing source-routing-path v4_R6 primary v4R2R6
set routing-options router-id 192.168.255.2
set routing-options autonomous-system 65540

```

R3

```

set system host-name R3
set system ports console log-out-on-disconnect
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 10.0.23.3/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 8
set interfaces ge-0/0/1 unit 0 family inet address 10.0.34.3/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 8
set interfaces lo0 unit 0 family inet address 192.168.255.3/32
set interfaces lo0 unit 0 family iso address 49.1921.6825.5003.00
set interfaces lo0 unit 0 family mpls maximum-labels 8
set policy-options policy-statement sspf term 1 from route-filter 192.168.255.3/32 exact
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3103
set policy-options policy-statement sspf term 1 then prefix-segment index 3003
set policy-options policy-statement sspf term 1 then prefix-segment node-segment
set policy-options policy-statement sspf term 1 then accept
set protocols isis interface all
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 64000
set protocols isis source-packet-routing node-segment ipv4-index 1003
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis level 2 wide-metrics-only
set protocols isis level 1 wide-metrics-only
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis export sspf
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
set protocols source-packet-routing
set routing-options router-id 192.168.255.3
set routing-options autonomous-system 65540

```

R4

```

set system host-name R4
set system ports console log-out-on-disconnect
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 10.0.34.4/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 8
set interfaces ge-0/0/1 unit 0 family inet address 10.0.47.4/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 8
set interfaces lo0 unit 0 family inet address 192.168.255.4/32
set interfaces lo0 unit 0 family iso address 49.1921.6825.5004.00
set interfaces lo0 unit 0 family mpls maximum-labels 8
set policy-options policy-statement sspf term 1 from route-filter 192.168.255.4/32 exact
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3104
set policy-options policy-statement sspf term 1 then prefix-segment index 3004
set policy-options policy-statement sspf term 1 then prefix-segment node-segment
set policy-options policy-statement sspf term 1 then accept
set protocols isis interface all
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 64000
set protocols isis source-packet-routing node-segment ipv4-index 1004
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis level 2 wide-metrics-only
set protocols isis level 1 wide-metrics-only
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis export sspf
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
set protocols source-packet-routing
set routing-options router-id 192.168.255.4
set routing-options autonomous-system 65540

```

R5

```

set system host-name R5
set system ports console log-out-on-disconnect
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 10.0.56.5/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 8
set interfaces ge-0/0/1 unit 0 family inet address 10.0.10.5/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 8
set interfaces lo0 unit 0 family inet address 192.168.255.5/32
set interfaces lo0 unit 0 family iso address 49.1921.6825.5005.00
set interfaces lo0 unit 0 family mpls maximum-labels 8
set policy-options policy-statement sspf term 1 from route-filter 192.168.255.5/32 exact
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3105
set policy-options policy-statement sspf term 1 then prefix-segment index 3005
set policy-options policy-statement sspf term 1 then prefix-segment node-segment
set policy-options policy-statement sspf term 1 then accept
set protocols isis interface all
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 64000
set protocols isis source-packet-routing node-segment ipv4-index 1005
set protocols isis source-packet-routing node-segment ipv6-index 2005
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis level 2 wide-metrics-only
set protocols isis level 1 wide-metrics-only
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis export sspf
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
set protocols source-packet-routing
set routing-options router-id 192.168.255.5
set routing-options autonomous-system 65540

```



R6

```
set system host-name R6
set system ports console log-out-on-disconnect
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 10.0.56.6/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 8
set interfaces ge-0/0/1 unit 0 family inet address 10.0.67.6/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 8
set interfaces lo0 unit 0 family inet address 192.168.255.6/32
set interfaces lo0 unit 0 family iso address 49.1921.6825.5006.00
set interfaces lo0 unit 0 family mpls maximum-labels 8
set policy-options policy-statement sspf term 1 from route-filter 192.168.255.6/32 exact
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3106
set policy-options policy-statement sspf term 1 then prefix-segment index 3006
set policy-options policy-statement sspf term 1 then prefix-segment node-segment
set policy-options policy-statement sspf term 1 then accept
set protocols isis interface all
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 64000
set protocols isis source-packet-routing node-segment ipv4-index 1006
set protocols isis source-packet-routing node-segment ipv6-index 2006
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis level 2 wide-metrics-only
set protocols isis level 1 wide-metrics-only
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis export sspf
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
set protocols source-packet-routing
set routing-options router-id 192.168.255.6
set routing-options autonomous-system 65540
```

R7

```
set system host-name R7
set system ports console log-out-on-disconnect
set chassis network-services enhanced-ip
set interfaces ge-0/0/0 unit 0 family inet address 10.0.47.7/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 8
set interfaces ge-0/0/1 unit 0 family inet address 10.0.67.7/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 8
set interfaces lo0 unit 0 family inet address 192.168.255.7/32
set interfaces lo0 unit 0 family iso address 49.1921.6825.5007.00
set interfaces lo0 unit 0 family mpls maximum-labels 8
set policy-options policy-statement sspf term 1 from route-filter 192.168.255.7/32 exact
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3107
set policy-options policy-statement sspf term 1 then prefix-segment index 3007
set policy-options policy-statement sspf term 1 then prefix-segment node-segment
set policy-options policy-statement sspf term 1 then accept
set protocols isis interface all
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 64000
set protocols isis source-packet-routing node-segment ipv4-index 1007
set protocols isis source-packet-routing node-segment ipv6-index 2007
set protocols isis source-packet-routing flex-algorithm 128
set protocols isis level 2 wide-metrics-only
set protocols isis level 1 wide-metrics-only
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
set protocols isis export sspf
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols rsvp interface all
set protocols rsvp interface fxp0.0 disable
set protocols source-packet-routing
set routing-options router-id 192.168.255.7
set routing-options autonomous-system 65540
```

### *Enable Default SIDs (Algorithm 0) in SPRING*

1. Configure the basic device settings such as hostname, IPv4 address, loopback interface address, NET address, family ISO, family MPLS (with maximum number of labels for segment routing routed paths), enhanced-ip mode, router-ID, and autonomous system (AS) number on all eight routers.

```

user@R0#
set chassis network-services enhanced-ip
set system host-name R0
set interfaces ge-0/0/0 unit 0 family inet address 10.0.10.10/24
set interfaces ge-0/0/0 unit 0 family iso
set interfaces ge-0/0/0 unit 0 family mpls maximum-labels 8
set interfaces ge-0/0/1 unit 0 family inet address 10.0.10.20/24
set interfaces ge-0/0/1 unit 0 family iso
set interfaces ge-0/0/1 unit 0 family mpls maximum-labels 8
set interfaces lo0 unit 0 family inet address 192.168.255.10/32
set interfaces lo0 unit 0 family iso address 49.1921.6825.5010.00
set interfaces lo0 unit 0 family mpls maximum-labels 8
set routing-options router-id 192.168.255.10
set routing-options autonomous-system 65540

```

2. Enable IS-IS, RSVP, and MPLS protocols on all interfaces of all eight devices. You can also specify trace files and operations for MPLS.

```

user@R0#
set protocols isis interface ge-0/0/0.0
set protocols isis interface ge-0/0/1.0
set protocols isis interface all
set protocols isis interface fxp0.0 disable
set protocols isis interface lo0.0 passive
set protocols mpls traceoptions file sspf-igp-short
set protocols mpls traceoptions file size 100m
set protocols mpls traceoptions file world-readable
set protocols mpls traceoptions flag ted-export
set protocols mpls traceoptions flag ted-import
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols rsvp interface all
deactivate protocols rsvp interface all

```

```
set protocols rsvp interface fxp0.0 disable
deactivate protocols rsvp interface fxp0.0 disable
```

3. Configure all routers to advertise their loopback address and specify the index and the node segment of the prefix segment.

```
user@R0#
set policy-options policy-statement sspf term 1 from route-filter 192.168.255.10/32 exact
set policy-options policy-statement sspf term 1 then prefix-segment index 3000
set policy-options policy-statement sspf term 1 then prefix-segment node-segment
set policy-options policy-statement sspf term 1 then accept
```

4. Configure the start-label and index-range of SRGB for SPRING. Configure the value of IPv4 node segment index and assign 128 flex algorithm.

```
user@R0#
set protocols isis source-packet-routing srgb start-label 400000
set protocols isis source-packet-routing srgb index-range 64000
set protocols isis source-packet-routing node-segment ipv4-index 1000
set protocols isis source-packet-routing flex-algorithm 128
```

5. Configure options for shortest-path-first (SPF) algorithm in IS-IS protocol to enable the source packet routing node segment labels for computing backup paths on R0, R1, and R2. Set maximum labels set to 8.

```
user@R0#
set protocols isis backup-spf-options use-post-convergence-lfa maximum-labels 8
set protocols isis backup-spf-options use-post-convergence-lfa maximum-backup-paths 8
set protocols isis backup-spf-options use-source-packet-routing
```

6. Configure traffic engineering options to choose label switched paths from spring-te and use the MPLS paths as next hops on R0, R1, and R2. Set the IS-IS export policy.

```
user@R0#
set protocols isis traffic-engineering tunnel-source-protocol spring-te
set protocols isis traffic-engineering family inet-mpls shortcuts
set protocols isis export sspf
```

7. Configure an R2 label 403002 (created for algorithm 0 to create default SPF SID) on R0 with R2 as its next hop to the destination R6 and enable use-for-shortcut. Create labels on R1, and R2 as well.

```
user@R0#
set protocols source-packet-routing segment-list v4R0R7 h1 label 403002
set protocols source-packet-routing source-routing-path V4_R7 use-for-shortcut
set protocols source-packet-routing source-routing-path V4_R7 to 192.168.255.6
set protocols source-packet-routing source-routing-path V4_R7 primary v4R0R7
```

```
user@R1#
set protocols source-packet-routing segment-list v4R1R2 h1 label 403006
set protocols source-packet-routing source-routing-path V4_R2 use-for-shortcut
set protocols source-packet-routing source-routing-path V4_R2 to 192.168.255.2
set protocols source-packet-routing source-routing-path V4_R2 primary v4R1R2
```

```
user@R2#
set protocols source-packet-routing segment-list v4R2R6 h1 label 403000
set protocols source-packet-routing segment-list v4R2R6 h2 label 403007
set protocols source-packet-routing source-routing-path v4_R6 use-for-shortcut
set protocols source-packet-routing source-routing-path v4_R6 to 192.168.255.2
set protocols source-packet-routing source-routing-path v4_R6 primary v4R2R6
```

8. Enter `commit` command to commit the configurations.

### ***Enable Strict SPF SIDs (Algorithm 1) in SPRING***

1. To replace the labels used for default SPF SIDs with labels to be used for strict SPF SIDs, configure the following:

```
user@R0#
delete protocols source-packet-routing segment-list v4R0R7 h1 label 403002
set protocols source-packet-routing segment-list v4R0R7 h1 label 403102
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3100
```

```
user@R1#
delete protocols source-packet-routing segment-list v4R1R2 h1 label 403006
```

```
set protocols source-packet-routing segment-list v4R1R2 h1 label 403106
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3101
```

```
user@R2#
delete protocols source-packet-routing segment-list v4R2R6 h1 label 403000
set protocols source-packet-routing segment-list v4R2R6 h1 label 403100
delete protocols source-packet-routing segment-list v4R2R6 h2 label 403007
set protocols source-packet-routing segment-list v4R2R6 h2 label 403107
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3102
```

## 2. Set/activate algorithm 1 on all other routers in the network.

```
user@R3#
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3103
```

```
user@R4#
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3104
```

```
user@R5#
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3105
```

```
user@R6#
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3106
```

```
user@R7#
set policy-options policy-statement sspf term 1 then prefix-segment algorithm 1 index 3107
```

## 3. Enter commit command to commit all configuration.

## Results

Check the results of the configuration:

```
user@R0# show
system {
    host-name R0;
    ports {
        console log-out-on-disconnect;
    }
}
chassis {
    network-services enhanced-ip;
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 10.0.10.10/24;
            }
            family iso;
            family mpls {
                maximum-labels 8;
            }
        }
    }
    ge-0/0/1 {
        unit 0 {
            family inet {
                address 10.0.10.20/24;
            }
            family iso;
            family mpls {
                maximum-labels 8;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.255.10/32;
            }
        }
    }
}
```

```

        family iso {
            address 49.1921.6825.5010.00;
        }
        family mpls {
            maximum-labels 8;
        }
    }
}

policy-options {
    policy-statement sspf {
        term 1 {
            from {
                route-filter 192.168.255.10/32 exact;
            }
            then {
                prefix-segment {
                    algorithm 1 index 3100;
                    index 3000;
                    node-segment;
                }
                accept;
            }
        }
    }
}

protocols {
    isis {
        interface ge-0/0/0.0;
        interface ge-0/0/1.0;
        interface all;
        interface fxp0.0 {
            disable;
        }
        interface lo0.0 {
            passive;
        }
        source-packet-routing {
            srgb start-label 400000 index-range 64000;
            node-segment ipv4-index 1000;
            flex-algorithm 128;
        }
        backup-spf-options {

```



```

        use-post-convergence-lfa {
            maximum-labels 8;
            maximum-backup-paths 8;
        }
        use-source-packet-routing;
    }
    traffic-engineering {
        tunnel-source-protocol {
            spring-te;
        }
        family inet-mpls {
            shortcuts;
        }
    }
    export sspf;
}
mpls {
    traceoptions {
        file sspf-igp-short size 100m world-readable;
        flag ted-export;
        flag ted-import;
    }
    interface all;
    interface fxp0.0 {
        disable;
    }
}
rsvp {
    inactive: interface all;
    interface fxp0.0 {
        inactive: disable;
    }
}
source-packet-routing {
    segment-list v4R0R7 {
        h1 label 403102;
    }
    source-routing-path V4_R7 {
        use-for-shortcut;
        to 192.168.255.6;
        primary {
            v4R0R7;
        }
    }
}

```

```
    }
  }
}
routing-options {
  router-id 192.168.255.10;
```

Verification

IN THIS SECTION

Verify IS-IS Adjacencies | 684

Verify Route Table inet.3 | 685

Verify Route Label (Default SPF) | 686

Verify Route Label (Strict SPF) | 687

Verify IS-IS Adjacencies

IN THIS SECTION

Purpose | 684

Action | 684

Meaning | 685

Purpose

Verify expected IS-IS adjacencies on the routing devices.

Action

From operational mode, enter the show isis adjacency command.

```
user@R0> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
ge-0/0/0.0     R1          1 Up          23 56:4:15:0:1c:d2
```

ge-0/0/0.0	R1	2	Up	25	56:4:15:0:1c:d2
ge-0/0/1.0	R5	1	Up	25	56:4:15:0:1c:eb
ge-0/0/1.0	R5	2	Up	24	56:4:15:0:1c:eb

**Meaning**

The output indicates that R0 has successfully formed IS-IS adjacencies on its `ge-0/0/0.0` and `ge-0/0/1.0` interfaces, which attach to their R1 and R5 routers, respectively.

**Verify Route Table `inet.3`**

IN THIS SECTION

● Purpose | 685

● Action | 685

● Meaning | 686

**Purpose**

Verify the `inet.3` routing table with the advertised.

**Action**

From operational mode, enter the `show route table inet.3` command.

```
regress@R0> show route table inet.3

inet.3: 7 destinations, 8 routes (7 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.255.1/32  *[*L-ISIS/14] 3d 19:43:17, metric 10
                  > to 10.0.10.1 via ge-0/0/0.0
192.168.255.2/32  *[*L-ISIS/14] 3d 19:43:17, metric 20
                  > to 10.0.10.1 via ge-0/0/0.0, Push 403002
192.168.255.3/32  *[*L-ISIS/14] 3d 19:43:17, metric 30
                  > to 10.0.10.1 via ge-0/0/0.0, Push 403003
192.168.255.4/32  *[*L-ISIS/14] 3d 19:43:17, metric 21
                  > to 10.0.10.1 via ge-0/0/0.0, Push 403004, Push 403002(top)
192.168.255.5/32  *[*L-ISIS/14] 3d 19:43:17, metric 10
```

```

> to 10.0.10.5 via ge-0/0/1.0
192.168.255.6/32 *[SPRING-TE/8] 3d 19:43:17, metric 1, metric2 20
> to 10.0.10.1 via ge-0/0/0.0, Push 403002
[L-ISIS/14] 3d 19:43:17, metric 1
> to 10.0.10.1 via ge-0/0/0.0, Push 403002
192.168.255.7/32 *[L-ISIS/14] 3d 19:43:17, metric 11
> to 10.0.10.1 via ge-0/0/0.0, Push 403007, Push 403002(top)

```

### Meaning

The output displays the routes on inet.3 table.

### Verify Route Label (Default SPF)

#### IN THIS SECTION

- Purpose | 686
- Action | 686
- Meaning | 687

### Purpose

Verify route labels created for default SPF on the routing devices.

### Action

From operational mode, enter the `show route label 403002` command.

```

user@R0> show route label 403002

mpls.0: 23 destinations, 23 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

403002          *[L-ISIS/14] 3d 20:17:24, metric 20
> to 10.0.10.1 via ge-0/0/0.0, Swap 403002

```

```

regress@R1> show route label 403002

```

```
mpls.0: 23 destinations, 23 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
403002          *[L-ISIS/14] 3d 20:31:53, metric 1
                > to 10.0.10.10 via ge-0/0/0.0, Push 403006
403002(S=0)     *[L-ISIS/14] 3d 20:31:53, metric 1
                > to 10.0.10.10 via ge-0/0/0.0, Push 403006
```

### **Meaning**

The output indicates that the packet is pushing R2's label 403002 to R1 to reach its next hop R2. But on R1, it picks up the tunnel B and pushes the label of its next hop R6- 403006, instead of getting forwarded from R1 to R2 on tunnel A.

### **Verify Route Label (Strict SPF)**

#### **IN THIS SECTION**

- [Purpose | 687](#)
- [Action | 687](#)
- [Meaning | 688](#)

### **Purpose**

Verify route labels created for strict SPF on the routing devices.

### **Action**

From operational mode, enter the show route label 403102 command.

```
user@R0> show route label 403102
```

```
mpls.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
403102          *[L-ISIS/14] 00:36:07, metric 20
                > to 10.0.10.1 via ge-0/0/0.0, Swap 403102
```

```
regress@R1> show route label 403102
```

```
mpls.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
```

```
+ = Active Route, - = Last Active, * = Both
```

```
403102          *[L-ISIS/14] 00:37:38, metric 10
                > to 10.0.12.2 via ge-0/0/1.0, Pop
403102(S=0)     *[L-ISIS/14] 00:37:38, metric 10
                > to 10.0.12.2 via ge-0/0/1.0, Pop
```

### ***Meaning***

The first output indicates that the packet with R2's label has reached R1. The second output indicates that the packet is now forwarded to R2 (on tunnel A), instead of getting picked by the tunnel B on R1. Once it reaches R2, it can complete the tunnel A path and reach R6.

# Configuring IS-IS Scaling and Throttling

## IN THIS CHAPTER

- [Understanding Link-State PDU Throttling for IS-IS Interfaces | 689](#)
- [Example: Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces | 690](#)
- [Understanding the Transmission Frequency for CSNPs on IS-IS Interfaces | 697](#)
- [Example: Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces | 698](#)
- [Understanding IS-IS Mesh Groups | 705](#)
- [Example: Configuring Mesh Groups of IS-IS Interfaces | 706](#)

## Understanding Link-State PDU Throttling for IS-IS Interfaces

Link-state PDU throttling by use of the `lsp-interval` statement is a mechanism to control the flooding pace to neighboring routing devices to prevent overloading them.

Control traffic (link-state PDU and related packets) might cause delays in user traffic (information packets) because control traffic always has precedence in terms of scheduling on the interface cards.

Unfortunately, the control traffic transmission rate does not get lower on low-bandwidth interfaces. Control traffic stays the same, regardless of line bandwidth.

Junos OS does not support automated calculation of link-state PDU throttling based on available bandwidth. It is assumed that even with link-state PDU pacing of 20 ms, the control traffic will not consume more than half of the interface bandwidth.

Thus, the `lsp-interval` statement helps to resolve two issues: regulating the control-traffic-to-user-traffic ratio, and protecting neighbors during transient situations.

The traffic subject to this pacing is non-self-originated traffic, which is traffic that has been originated by other routers, not the local router. Junos OS has hard-coded rate limiting for locally generated link-state PDUs. All the link-state PDUs are paced using a 20 ms timer. Additionally, there is logic that makes sure that the adjacency is reliably up for some time before advertising the adjacency.

## RELATED DOCUMENTATION

| [Example: Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces](#) | 690

# Example: Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces

## IN THIS SECTION

- [Requirements](#) | 690
- [Overview](#) | 690
- [Configuration](#) | 691
- [Verification](#) | 694

This example shows how to modify the link-state PDU interval time.

## Requirements

Before you begin, configure IS-IS. See ["Example: Configuring IS-IS" on page 15](#) for information about the sample IS-IS configuration.

## Overview

To keep reachability information in the network current, link-state protocols need to originate, distribute, and revoke or time-out topology information. In IS-IS, topology information is encoded in link-state PDUs.

By default, the routing device sends one link-state PDU out an interface every 100 milliseconds. To modify this interval, include the `lsp-interval` statement:

```
lsp-interval milliseconds;
```

To disable the transmission of all link-state PDUs, set the interval to 0.

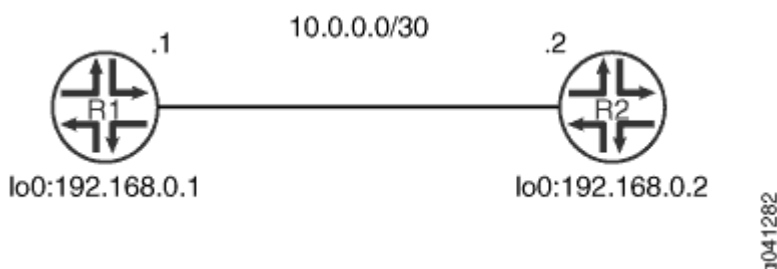
Link-state PDU throttling by use of the `lsp-interval` statement controls the flooding pace to neighboring routing devices in order to not overload them and also to ensure that user traffic is not delayed on low-bandwidth links.



In this example, an IS-IS routing device on a LAN segment is configured to send link-state PDUs every 1000 milliseconds.

Figure 46 on page 691 shows the topology used in this example.

Figure 46: IS-IS Link-State PDU Interval Topology



This example describes the steps on Device R1.

## Configuration

### IN THIS SECTION

- Procedure | 691

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

### Device R1

```
set interfaces fe-1/2/0 unit 0 description to-R2
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.1/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set protocols isis traceoptions file isis-trace
```

```

set protocols isis traceoptions flag lsp
set protocols isis interface fe-1/2/0.0 lsp-interval 1000
set protocols isis interface lo0.0

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure the link-state PDU interval:

1. Configure the interfaces.

```

[edit interfaces]
user@R1# set fe-1/2/0 unit 0 description to-R2
user@R1# set fe-1/2/0 unit 0 family inet address 10.0.0.1/30
user@R1# set fe-1/2/0 unit 0 family iso
user@R1# set lo0 unit 0 family inet address 192.168.0.1/32
user@R1# set lo0 unit 0 family iso address 49.0002.0192.0168.0001.00

```

2. Enable IS-IS on the interfaces.

```

[edit protocols isis]
user@R1# set interface fe-1/2/0.0
user@R1# set interface lo0.0

```

3. Modify the link-state PDU interval.

```

[edit protocols isis interface fe-1/2/0.0]
user@R1# set lsp-interval 1000

```

4. (Optional) Enable tracing for tracking link-state PDU operations.

```

[edit protocols isis traceoptions]
user@R1# set file isis-trace
user@R1# set flag lsp

```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces` and `show protocols` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces
fe-1/2/0 {
  unit 0 {
    description to-R2;
    family inet {
      address 10.0.0.1/30;
    }
    family iso;
  }
}
lo0 {
  unit 0 {
    family inet {
      address 192.168.0.1/32;
    }
    family iso {
      address 49.0002.0192.0168.0001.00;
    }
  }
}
```

```
user@R1# show protocols
isis {
  traceoptions {
    file isis-trace;
    flag lsp;
  }
  interface fe-1/2/0.0 {
    lsp-interval 1000;
  }
  interface lo0.0;
}
```

If you are done configuring the device, enter `commit` from configuration mode. Repeat the configuration on Device R2.

## Verification

### IN THIS SECTION

- [Verifying the Link-State PDU Interval | 694](#)
- [Checking the Link-State PDU Statistics | 695](#)
- [Checking the Trace Log | 696](#)

Confirm that the configuration is working properly.

### Verifying the Link-State PDU Interval

#### Purpose

Check the link-state PDU interval setting on the IS-IS interface.

#### Action

From operational mode, enter the `show isis interface extensive` command.

```
user@R1> show isis interface extensive
fe-1/2/0.0
  Index: 70, State: 0x6, Circuit id: 0x1, Circuit type: 3
  LSP interval: 1000 ms, CSNP interval: 10 s, Loose Hello padding
  Adjacency advertisement: Advertise
  Level 1
    Adjacencies: 1, Priority: 64, Metric: 10
    Hello Interval: 9.000 s, Hold Time: 27 s
    Designated Router: R2.02 (not us)
  Level 2
    Adjacencies: 1, Priority: 64, Metric: 10
    Hello Interval: 9.000 s, Hold Time: 27 s
    Designated Router: R2.02 (not us)
```

#### Meaning

The output shows that the link-state PDU interval is set to 1000 milliseconds.

## Checking the Link-State PDU Statistics

### Purpose

Check the number of link-state PDUs sent and received.

### Action

From operational mode, enter the `show isis statistics` command.

```
user@R1> show isis statistics
```

IS-IS statistics for R1:

PDU type	Received	Processed	Drops	Sent	Rexmit
<b>LSP</b>	<b>24</b>	<b>24</b>	<b>0</b>	<b>13</b>	<b>0</b>
IIH	2467	24	0	836	0
CSNP	474	474	0	0	0
PSNP	0	0	0	0	0
Unknown	0	0	0	0	0
Totals	2965	522	0	849	0

Total packets received: 2965 Sent: 849

SNP queue length: 0 Drops: 0

LSP queue length: 0 Drops: 0

SPF runs: 14

Fragments rebuilt: 15

LSP regenerations: 6

Purges initiated: 0

```
user@R2> show isis statistics
```

IS-IS statistics for R2:

PDU type	Received	Processed	Drops	Sent	Rexmit
<b>LSP</b>	<b>13</b>	<b>13</b>	<b>0</b>	<b>24</b>	<b>0</b>
IIH	828	15	0	2459	0
CSNP	0	0	0	474	0
PSNP	0	0	0	0	0
Unknown	0	0	0	0	0
Totals	841	28	0	2957	0

```
Total packets received: 841 Sent: 2957
```

```
SNP queue length: 0 Drops: 0
```

```
LSP queue length: 0 Drops: 0
```

```
SPF runs: 17
```

```
Fragments rebuilt: 26
```

```
LSP regenerations: 11
```

```
Purges initiated: 0
```

## Meaning

The output shows the number of link-state PDUs sent and received on Device R1 and Device R2.

## Checking the Trace Log

### Purpose

Check the IS-IS trace log to view the interval between packets.

### Action

From operational mode, enter the `show log isis | match lsp` command.

```
user@R1> show log isis | match lsp
```

```
Jun 18 15:27:02.692031 Received L1 LSP R2.00-00, on interface fe-1/2/0.0
```

```
Jun 18 15:27:02.692753 Updating L1 LSP R2.00-00 in TED
```

```
Jun 18 15:27:44.396480 Updating L1 LSP R1.00-00 in TED
```

```
Jun 18 15:27:45.398077 Sending L1 LSP R1.00-00 on interface fe-1/2/0.0
```

```
Jun 18 15:28:44.689024 Received L1 LSP R2.02-00, on interface fe-1/2/0.0
```

```
Jun 18 15:28:44.689663 Updating L1 LSP R2.02-00 in TED
```

```
Jun 18 15:29:15.954900 Updating L2 LSP R1.00-00 in TED
```

```
Jun 18 15:29:16.955620 Sending L2 LSP R1.00-00 on interface fe-1/2/0.0
```

```
Jun 18 15:29:28.789986 Received L2 LSP R2.00-00, on interface fe-1/2/0.0
```

```
Jun 18 15:29:28.790620 Updating L2 LSP R2.00-00 in TED
```

```
Jun 18 15:30:27.727892 Received L2 LSP R2.02-00, on interface fe-1/2/0.0
```

```
Jun 18 15:30:27.728519 Updating L2 LSP R2.02-00 in TED
```

## Meaning

The output shows that Level 1 and Level 2 link-state PDUs are being sent and received roughly every 1000 milliseconds (1 second).

## RELATED DOCUMENTATION

[Understanding Link-State PDU Throttling for IS-IS Interfaces | 689](#)

[Example: Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces | 698](#)

## Understanding the Transmission Frequency for CSNPs on IS-IS Interfaces

The complete sequence number PDU (CSNP) interval controls the frequency at which a routing device sends a directory of its link-state database.

When IS-IS is activated on a routing device's interface, the device first sends some IS-IS hello packets (IIHs) to its neighbors to ensure that the circuit is capable of transporting packets in both directions. In the IIHs, the router embeds information about the designated router (also called the designated intermediate system or DIS). One of the designated router roles on an IS-IS broadcast circuit is to synchronize the link-state databases on LANs. The designated router does this by periodically sending a directory of its link-state database, which is received by all the routing devices on a LAN.

If the routing device is the designated router on a LAN, IS-IS sends CSNPs every 10 seconds. If the routing device is on a point-to-point interface, it sends CSNPs every 5 seconds. The general recommendation is to use the default values or to increase the CSNP interval if there are a large number of broadcast circuits that need to be supplied with fresh CSNPs. Increasing the interval can help protect against CSNP flooding.

## RELATED DOCUMENTATION

[Example: Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces | 698](#)

## Example: Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces

### IN THIS SECTION

- Requirements | 698
- Overview | 698
- Configuration | 699
- Verification | 702

This example shows how to modify the complete sequence number PDU (CSNP) interval on IS-IS interfaces.

### Requirements

Before you begin, configure IS-IS. See ["Example: Configuring IS-IS" on page 15](#) for information about the sample IS-IS configuration.

### Overview

CSNPs contain a complete list of all link-state PDUs in the IS-IS database. CSNPs are sent periodically on all links, and the receiving systems use the information in the CSNP to update and synchronize their link-state PDU databases. The designated router multicasts CSNPs on broadcast links in place of sending explicit acknowledgments for each link-state PDU.

If the routing device is the designated router on a LAN, IS-IS sends CSNPs every 10 seconds. You might want to modify the default interval to protect against CSNP flooding.



**NOTE:** The **csnp-interval** statement is effective only when configured on LAN interfaces.

To modify the CSNP interval, include the **csnp-interval** statement:

```
csnp-interval seconds;
```

The time can range from 1 through 65,535 seconds.



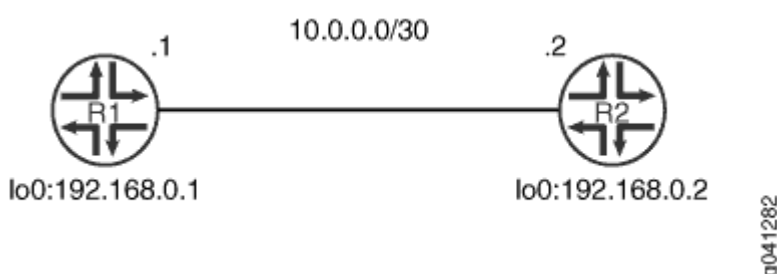
To configure the interface not to send any CSNPs, specify the disable option:

```
cnp-interval disable;
```

In this example, an IS-IS routing device on a LAN segment is configured to send CSNPs every 30 seconds.

Figure 47 on page 699 shows the topology used in this example.

**Figure 47: IS-IS CSNP Interval Topology**



This example describes the steps on Device R1.

## Configuration

### IN THIS SECTION

- Procedure | [699](#)

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R1

```
set interfaces fe-1/2/0 unit 0 description to-R2
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.1/30
```

```

set interfaces fe-1/2/0 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set protocols isis traceoptions file isis-trace
set protocols isis traceoptions flag csnp
set protocols isis interface fe-1/2/0.0 csnp-interval 30
set protocols isis interface lo0.0

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure the CSNP interval:

1. Configure the interfaces.

```

[edit interfaces]
user@R1# set fe-1/2/0 unit 0 description to-R2
user@R1# set fe-1/2/0 unit 0 family inet address 10.0.0.1/30
user@R1# set fe-1/2/0 unit 0 family iso
user@R1# set lo0 unit 0 family inet address 192.168.0.1/32
user@R1# set lo0 unit 0 family iso address 49.0002.0192.0168.0001.00

```

2. Enable IS-IS on the interfaces.

```

[edit protocols isis]
user@R1# set interface fe-1/2/0.0
user@R1# set interface lo0.0

```

3. Modify the CSNP interval.

```

[edit protocols isis interface fe-1/2/0.0]
user@R1# set csnp-interval 30

```

#### 4. (Optional) Enable tracing for tracking CSNP operations.

```
[edit protocols isis traceoptions]  
user@R1# set file isis-trace  
user@R1# set flag csn
```

### Results

From configuration mode, confirm your configuration by entering the `show interfaces` and `show protocols` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces  
fe-1/2/0 {  
  unit 0 {  
    description to-R2;  
    family inet {  
      address 10.0.0.1/30;  
    }  
    family iso;  
  }  
}  
lo0 {  
  unit 0 {  
    family inet {  
      address 192.168.0.1/32;  
    }  
    family iso {  
      address 49.0002.0192.0168.0001.00;  
    }  
  }  
}
```

```
user@R1# show protocols  
isis {  
  traceoptions {  
    file isis-trace;  
    flag csn;  
  }  
}
```

```

interface fe-1/2/0.0 {
    csnp-interval 30;
}
interface lo0.0;
}

```

If you are done configuring the device, enter `commit` from configuration mode. Repeat the configuration on Device R2.

## Verification

### IN THIS SECTION

- [Verifying the CSNP Interval | 702](#)
- [Checking the CSNP Statistics | 703](#)
- [Checking the IS-IS Log | 704](#)

Confirm that the configuration is working properly.

### Verifying the CSNP Interval

#### Purpose

Check the CSNP interval setting on the IS-IS interface.

#### Action

From operational mode, enter the `show isis interface extensive` command.

```

user@R1> show isis interface extensive
IS-IS interface database:
fe-1/2/0.0
  Index: 70, State: 0x6, Circuit id: 0x1, Circuit type: 3
  LSP interval: 100 ms, CSNP interval: 30 s, Loose Hello padding
  Adjacency advertisement: Advertise
  Level 1
    Adjacencies: 1, Priority: 64, Metric: 10
    Hello Interval: 9.000 s, Hold Time: 27 s

```

```

    Designated Router: R2.02 (not us)
Level 2
    Adjacencies: 1, Priority: 64, Metric: 10
    Hello Interval: 9.000 s, Hold Time: 27 s
    Designated Router: R2.02 (not us)

```

## Meaning

The output shows that the CSNP interval is set to 30 seconds.

## Checking the CSNP Statistics

### Purpose

Checking the number of CSNPs sent and received.

### Action

From operational mode, enter the `show isis statistics` command.

```
user@R1> show isis statistics
```

IS-IS statistics for R1:

PDU type	Received	Processed	Drops	Sent	Rexmit
LSP	5	5	0	3	0
IIH	94	20	0	43	0
<b>CSNP</b>	<b>6</b>	<b>6</b>	<b>0</b>	<b>0</b>	<b>0</b>
PSNP	0	0	0	0	0
Unknown	0	0	0	0	0
Totals	105	31	0	46	0

Total packets received: 105 Sent: 46

SNP queue length: 0 Drops: 0

LSP queue length: 0 Drops: 0

SPF runs: 5

Fragments rebuilt: 5

LSP regenerations: 0

Purges initiated: 0

-----

user@R2> **show isis statistics**

IS-IS statistics for R2:

PDU type	Received	Processed	Drops	Sent	Rexmit
LSP	3	3	0	5	0
IIH	35	11	0	86	0
<b>CSNP</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>6</b>	<b>0</b>
PSNP	0	0	0	0	0
Unknown	0	0	0	0	0
Totals	38	14	0	97	0

Total packets received: 38 Sent: 97

SNP queue length: 0 Drops: 0

LSP queue length: 0 Drops: 0

SPF runs: 7

Fragments rebuilt: 7

LSP regenerations: 0

Purges initiated: 0

## Meaning

The output shows the number of CSNPs sent and received on Device R1 and Device R2.



**NOTE:** On broadcast links, only the designated intermediate system (DIS) sends CSNPs.

## Checking the IS-IS Log

### Purpose

Check the IS-IS trace log to view the interval between packets.

## Action

From operational mode, enter the `show log isis-trace | match csn` command.

```
user@R1> show log isis-trace | match csn

Jun 18 14:36:19.504064 Received L1 CSN, source R2, interface fe-1/2/0.0
Jun 18 14:36:19.523065 Received L2 CSN, source R2, interface fe-1/2/0.0
Jun 18 14:36:48.904120 Received L1 CSN, source R2, interface fe-1/2/0.0
Jun 18 14:36:48.916425 Received L2 CSN, source R2, interface fe-1/2/0.0
Jun 18 14:37:14.954447 Received L1 CSN, source R2, interface fe-1/2/0.0
Jun 18 14:37:14.971329 Received L2 CSN, source R2, interface fe-1/2/0.0
Jun 18 14:37:44.227106 Received L1 CSN, source R2, interface fe-1/2/0.0
Jun 18 14:37:44.244181 Received L2 CSN, source R2, interface fe-1/2/0.0
```

## Meaning

The output shows that Level 1 and Level 2 CSNPs are being received roughly every 30 seconds.

## RELATED DOCUMENTATION

[Understanding the Transmission Frequency for CSNPs on IS-IS Interfaces | 697](#)

[Example: Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces | 690](#)

## Understanding IS-IS Mesh Groups

A *mesh group* is a set of routing devices that are fully connected. That is, they have a fully meshed topology.

Junos OS supports IS-IS mesh groups as documented in RFC 2973, *IS-IS Mesh Groups*.

When link-state PDUs are being flooded throughout an area, each router within a mesh group receives only a single copy of a link-state PDU instead of receiving one copy from each neighbor, thus minimizing the overhead associated with the flooding of link-state PDUs.

Mesh groups provide a scaling method for the flooding subsystem. In enterprise networks, dense flooding graphs that, for example, lease a Layer 2 VPN service (an overlay network) to fully mesh its WAN routers might continue to be a useful architecture. In such cases, mesh groups might be useful.

## RELATED DOCUMENTATION

[Example: Configuring Mesh Groups of IS-IS Interfaces | 706](#)

## Example: Configuring Mesh Groups of IS-IS Interfaces

### IN THIS SECTION

- [Requirements | 706](#)
- [Overview | 706](#)
- [Configuration | 707](#)
- [Verification | 711](#)

This example shows how to configure mesh groups of IS-IS interfaces.

### Requirements

No special configuration beyond device initialization is required before configuring this example.

### Overview

When link-state PDUs are being flooded throughout an area, each router within a mesh group receives only a single copy of a link-state PDU instead of receiving one copy from each neighbor, thus minimizing the overhead associated with the flooding of link-state PDUs.

To create a mesh group and designate that an interface be part of the group, assign a mesh-group number to all the routing device interfaces in the group:

```
mesh-group value;
```

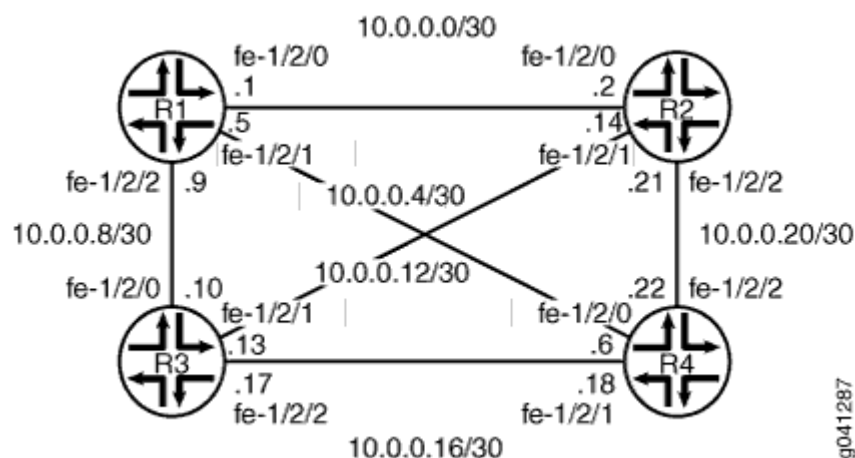
To prevent an interface in the mesh group from flooding link-state PDUs, configure blocking on that interface:

```
mesh-group blocked;
```

[Figure 48 on page 707](#) shows the topology used in this example.



Figure 48: IS-IS Mesh Topology



"CLI Quick Configuration" on page 707 shows the configuration for all of the devices in Figure 48 on page 707. The section "No Link Title" on page 709 describes the steps on Device R1.

## Configuration

### IN THIS SECTION

- Procedure | 707

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

#### Device R1

```
set interfaces fe-1/2/0 unit 0 description to-R2
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.1/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R4
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.5/30
set interfaces fe-1/2/1 unit 0 family iso
```

```

set interfaces fe-1/2/2 unit 0 description to-R3
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.9/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set protocols isis interface fe-1/2/0.0 mesh-group 1
set protocols isis interface fe-1/2/1.0 mesh-group 1
set protocols isis interface fe-1/2/2.0 mesh-group 1
set protocols isis interface lo0.0

```

## Device R2

```

set interfaces fe-1/2/0 unit 0 description to-R1
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.2/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R1
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.14/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/2 unit 0 description to-R1
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.21/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0002.00
set protocols isis interface fe-1/2/0.2 mesh-group 1
set protocols isis interface fe-1/2/1.0 mesh-group 1
set protocols isis interface fe-1/2/2.0 mesh-group 1
set protocols isis interface lo0.0

```

## Device R3

```

set interfaces fe-1/2/0 unit 0 description to-R1
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.10/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R2
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.13/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/2 unit 0 description to-R4
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.17/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0003.00

```

```

set protocols isis interface fe-1/2/0.0 mesh-group 1
set protocols isis interface fe-1/2/1.0 mesh-group 1
set protocols isis interface fe-1/2/2.0 mesh-group 1
set protocols isis interface lo0.0

```

## Device R4

```

set interfaces fe-1/2/0 unit 0 description to-R1
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.6/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/1 unit 0 description to-R3
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.18/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/2 unit 0 description to-R2
set interfaces fe-1/2/2 unit 0 family inet address 10.0.0.22/30
set interfaces fe-1/2/2 unit 0 family iso
set interfaces lo0 unit 0 family inet address 192.168.0.4/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0004.00
set protocols isis interface fe-1/2/0.0 mesh-group 1
set protocols isis interface fe-1/2/1.0 mesh-group 1
set protocols isis interface fe-1/2/2.0 mesh-group 1
set protocols isis interface lo0.0

```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure an IS-IS mesh group:

1. Configure the interfaces.

```

[edit interfaces]
user@R1# set fe-1/2/0 unit 0 description to-R2
user@R1# set fe-1/2/0 unit 0 family inet address 10.0.0.1/30
user@R1# set fe-1/2/0 unit 0 family iso
user@R1# set fe-1/2/1 unit 0 description to-R4
user@R1# set fe-1/2/1 unit 0 family inet address 10.0.0.5/30
user@R1# set fe-1/2/1 unit 0 family iso
user@R1# set fe-1/2/2 unit 0 description to-R3

```

```

user@R1# set fe-1/2/2 unit 0 family inet address 10.0.0.9/30
user@R1# set fe-1/2/2 unit 0 family iso
user@R1# set lo0 unit 0 family inet address 192.168.0.1/32
user@R1# set lo0 unit 0 family iso address 49.0002.0192.0168.0001.00

```

2. Enable IS-IS on the interfaces, and assign a mesh group number.

```

[edit protocols isis]
user@R1# set interface fe-1/2/0.0 mesh-group 1
user@R1# set interface fe-1/2/1.0 mesh-group 1
user@R1# set interface fe-1/2/2.0 mesh-group 1
user@R1# set interface lo0.0

```

## Results

From configuration mode, confirm your configuration by entering the `show interfaces` and `show protocols` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```

user@R1# show interfaces
fe-1/2/0 {
    unit 0 {
        description to-R2;
        family inet {
            address 10.0.0.1/30;
        }
        family iso;
    }
}
fe-1/2/1 {
    unit 0 {
        description to-R4;
        family inet {
            address 10.0.0.5/30;
        }
        family iso;
    }
}
fe-1/2/2 {
    unit 0 {

```

```

        description to-R3;
        family inet {
            address 10.0.0.9/30;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.0.1/32;
        }
        family iso {
            address 49.0002.0192.0168.0001.00;
        }
    }
}
}

```

```

user@R1# show protocols
isis {
    interface fe-1/2/0.0 {
        mesh-group 1;
    }
    interface fe-1/2/1.0 {
        mesh-group 1;
    }
    interface fe-1/2/2.0 {
        mesh-group 1;
    }
    interface lo0.0;
}

```

If you are done configuring the device, enter **commit** from configuration mode.

## Verification

### IN THIS SECTION

● [Checking the Interface Mesh Group | 712](#)

- [Checking the IS-IS Statistics | 713](#)

Confirm that the configuration is working properly.

## Checking the Interface Mesh Group

### Purpose

Verify that the mesh group is enabled on the IS-IS interfaces.

### Action

From operational mode, enter the `show isis interface extensive` command.

```
user@R1> show isis interface extensive

IS-IS interface database:
lo0.0
  Index: 68, State: 0x6, Circuit id: 0x1, Circuit type: 0
  LSP interval: 100 ms, CSNP interval: disabled, Loose Hello padding
  Adjacency advertisement: Advertise
  Level 1
    Adjacencies: 0, Priority: 64, Metric: 0
    Passive
  Level 2
    Adjacencies: 0, Priority: 64, Metric: 0
    Passive
fe-1/2/0.1
  Index: 73, State: 0x206, Circuit id: 0x1, Circuit type: 3
  LSP interval: 100 ms, CSNP interval: 10 s, Loose Hello padding
  Adjacency advertisement: Advertise
  CSNP interval: disabled, Mesh group: 1
  Level 1
    Adjacencies: 1, Priority: 64, Metric: 10
    Hello Interval: 9.000 s, Hold Time: 27 s
    Designated Router: tp5-R2.03 (not us)
  Level 2
    Adjacencies: 1, Priority: 64, Metric: 10
```

```

    Hello Interval: 9.000 s, Hold Time: 27 s
    Designated Router: tp5-R2.03 (not us)
fe-1/2/1.0
  Index: 75, State: 0x206, Circuit id: 0x1, Circuit type: 3
  LSP interval: 100 ms, CSNP interval: 10 s, Loose Hello padding
  Adjacency advertisement: Advertise
  CSNP interval: disabled, Mesh group: 1
Level 1
  Adjacencies: 1, Priority: 64, Metric: 10
  Hello Interval: 9.000 s, Hold Time: 27 s
  Designated Router: tp5-R4.03 (not us)
Level 2
  Adjacencies: 1, Priority: 64, Metric: 10
  Hello Interval: 9.000 s, Hold Time: 27 s
  Designated Router: tp5-R4.03 (not us)
fe-1/2/2.0
  Index: 76, State: 0x206, Circuit id: 0x1, Circuit type: 3
  LSP interval: 100 ms, CSNP interval: 10 s, Loose Hello padding
  Adjacency advertisement: Advertise
  CSNP interval: disabled, Mesh group: 1
Level 1
  Adjacencies: 1, Priority: 64, Metric: 10
  Hello Interval: 9.000 s, Hold Time: 27 s
  Designated Router: tp5-R3.02 (not us)
Level 2
  Adjacencies: 1, Priority: 64, Metric: 10
  Hello Interval: 9.000 s, Hold Time: 27 s
  Designated Router: tp5-R3.02 (not us)

```

## Meaning

Mesh group: 1 in the output shows that the mesh group is enabled as expected.

## Checking the IS-IS Statistics

### Purpose

Verify that the number of link-state PDUs received and sent is less than what it would be if the mesh group were not enabled.

## Action

From operational mode, enter the `show isis statistics` command.

```
user@R1> show isis statistics
```

IS-IS statistics for tp5-R1:

PDU type	Received	Processed	Drops	Sent	Rexmit
<b>LSP</b>	<b>73</b>	<b>73</b>	<b>0</b>	<b>37</b>	<b>0</b>
IIH	4463	85	0	1525	0
CSNP	1294	1294	0	0	0
PSNP	0	0	0	2	0
Unknown	0	0	0	0	0
Totals	5830	1452	0	1564	0

Total packets received: 5830 Sent: 1564

SNP queue length: 0 Drops: 0

LSP queue length: 0 Drops: 0

SPF runs: 26

Fragments rebuilt: 15

LSP regenerations: 4

Purges initiated: 0

## Meaning

After the adjacencies have been up for about 38 minutes, the output shows that Device R1 has received 73 link-state PDUs and sent 37 link-state PDUs. In the same topology in the same amount of time without the mesh group enabled, Device R1 would have received roughly 156 link-state PDUs and sent roughly 117 link-state PDUs.

## RELATED DOCUMENTATION

[Understanding IS-IS Mesh Groups](#) | 705



# Configuring IS-IS CLNS

## IN THIS CHAPTER

- [Understanding IS-IS for CLNS | 715](#)
- [Example: Configuring IS-IS for CLNS | 715](#)

## Understanding IS-IS for CLNS

IS-IS extensions provide the basic interior gateway protocol (IGP) support for collecting intradomain routing information for Connectionless Network Service (CLNS) destinations within a CLNS network. Routers that learn host addresses through End System-to-Intermediate System (ES-IS) can advertise the addresses to other routers (intermediate systems) by using IS-IS.

For more information about IS-IS, see the ISO 10589 standard.

## RELATED DOCUMENTATION

[CLNS Overview](#)

[Example: Configuring IS-IS for CLNS | 715](#)

## Example: Configuring IS-IS for CLNS

## IN THIS SECTION

- [Requirements | 716](#)
- [Overview | 716](#)
- [Configuration | 716](#)

- [Verification | 718](#)

This example shows how to create a routing instance and enable IS-IS protocol on all interfaces.

## Requirements

Before you begin, configure the network interfaces. See [Interfaces User Guide for Security Devices](#).

## Overview

The configuration instructions in this topic describe how to create a routing-instance called `aaaa`, enable IS-IS on all interfaces, and define BGP export policy name (`dist-bgp`), family (ISO), and protocol (BP), and apply the export policy to IS-IS.

## Configuration

### IN THIS SECTION

- [Procedure | 716](#)

## Procedure

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
set routing-instances aaaa protocols isis clns-routing
set routing-instances aaaa protocols isis interface all
set routing-instances aaaa protocols isis no-ipv4-routing no-ipv6-routing
set policy-options policy-statement dist-bgp from family iso protocol bgp
set policy-options policy-statement dist-bgp then accept
set routing-instances aaaa protocols isis export dist-bgp
```

## Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure IS-IS for CLNS:

1. Configure the routing instance.

```
[edit]
user@host# edit routing-instances aaaa
```

2. Enable CLNS routing.

```
[edit routing-instances aaaa]
user@host# set protocols isis clns-routing
```

3. Enable IS-IS on all interfaces.

```
[edit routing-instances aaaa]
user@host# set protocols isis interface all
```

4. (Optional) Disable IPv4 and IPv6 routing to configure a pure CLNS network .

```
[edit routing-instances aaaa]
user@host# set protocols isis no-ipv4-routing no-ipv6-routing
```

5. Define the BGP export policy name, family, and protocol.

```
[edit policy-options]
user@host# set policy-statement dist-bgp from family iso protocol bgp
```

6. Define the action for the export policy.

```
[edit policy-options]
user@host# set policy-statement dist-bgp then accept
```

## 7. Apply the export policy to IS-IS.

```
[edit routing-instances aaaa]
user@host# set protocols isis export dist-bgp
```

## Results

From configuration mode, confirm your configuration by entering the `show routing-instances` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show routing-instances
aaaa {
  protocols {
    isis {
      export dist-bgp;
      no-ipv4-routing;
      no-ipv6-routing;
      clns-routing;
      interface all;
    }
  }
}
```

If you are done configuring the device, enter `commit` from configuration mode.

## Verification

### IN THIS SECTION

- [Verifying Routing-Instance for CLNS | 719](#)
- [Verifying IS-IS for CLNS | 719](#)

Confirm that the configuration is working properly.

## Verifying Routing-Instance for CLNS

### Purpose

Verify that the policy options are enabled for the routing instance.

### Action

From operational mode, enter the `show routing-instances` command.

## Verifying IS-IS for CLNS

### Purpose

Verify that IS-IS is enabled.

### Action

From operational mode, enter the `show protocols` command.

## RELATED DOCUMENTATION

---

[CLNS Configuration Overview](#)

---

[Understanding IS-IS for CLNS | 715](#)

---

[Verifying a CLNS VPN Configuration](#)

# Configuring IS-IS on Logical Systems

## IN THIS CHAPTER

- Introduction to Logical Systems | 720
- Example: Configuring IS-IS on Logical Systems Within the Same Router | 721
- Example: Configuring an IS-IS Default Route Policy on Logical Systems | 736

## Introduction to Logical Systems

For many years, engineers have combined power supplies, routing hardware and software, forwarding hardware and software, and physical interfaces into a networking device known as a router. Networking vendors have created large routers and small routers, but all routers have been placed into service as individual devices. As a result, the router has been considered a single physical device for most of its history.

The concept of *logical systems* breaks with this tradition. With the Junos® operating system (Junos OS), you can partition a single router into multiple logical devices that perform independent routing tasks. Because logical systems perform a subset of the tasks once handled by the *main router*, logical systems offer an effective way to maximize the use of a single routing or switching platform.

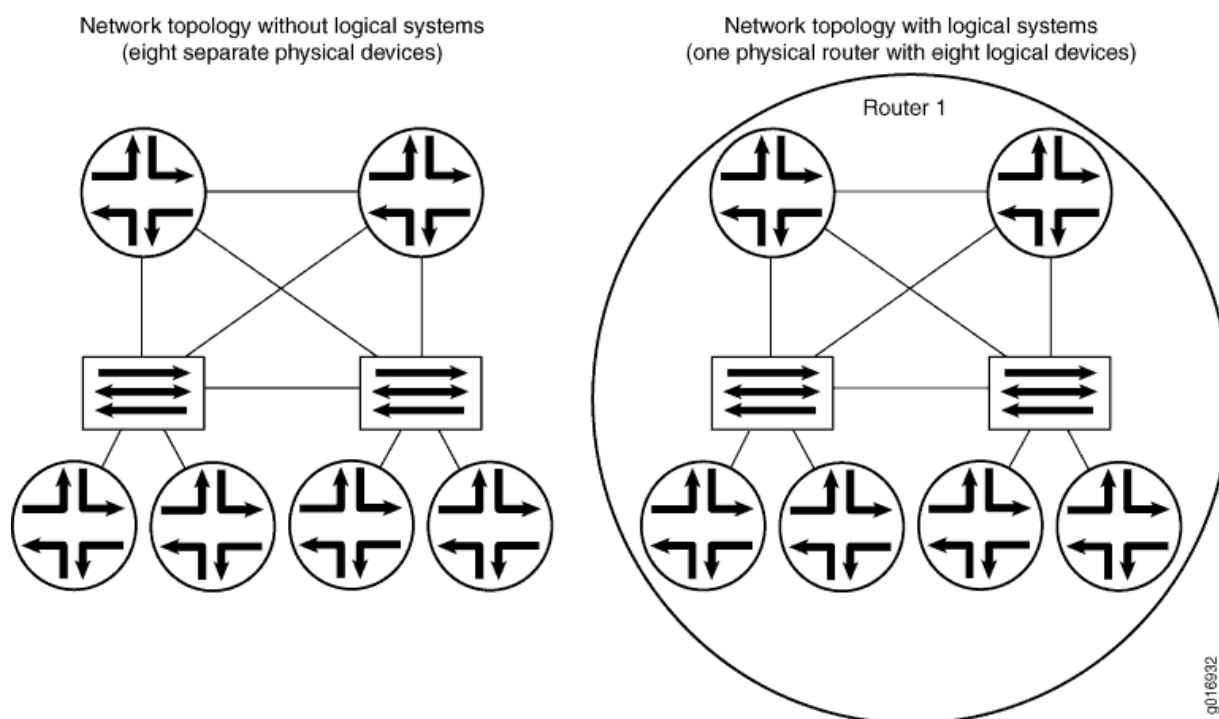


**NOTE:** All configuration statements, operational commands, `show` command output, error messages, log messages, and SNMP MIB objects that contain the string `logical-router` have been changed to `logical-system`.

Traditionally, service provider network design requires multiple layers of switches and routers. These devices transport packet traffic between customers. As seen on the left side of [Figure 49 on page 721](#), access devices are connected to edge devices, which are in turn connected to core devices.

However, this complexity can lead to challenges in maintenance, configuration, and operation. To reduce such complexity, Juniper Networks supports logical systems. Logical systems perform a subset of the actions of the main router and have their own unique routing tables, interfaces, policies, and routing instances. As shown on the right side of [Figure 49 on page 721](#), a set of logical systems within a single router can handle the functions previously performed by several small routers.

Figure 49: Logical Systems Concepts



## Example: Configuring IS-IS on Logical Systems Within the Same Router

### IN THIS SECTION

- [Requirements | 722](#)
- [Overview | 722](#)
- [Configuration | 723](#)
- [Verification | 731](#)

This example shows how to configure an IS-IS network by using multiple logical systems that are running on a single physical router. The logical systems are connected by logical tunnel interfaces.

## Requirements

You must connect the logical systems by using logical tunnel (lt) interfaces. See *Example: Connecting Logical Systems Within the Same Device Using Logical Tunnel Interfaces on MX Series Routers and EX Series Switches*.

## Overview

### IN THIS SECTION

- [Topology | 722](#)

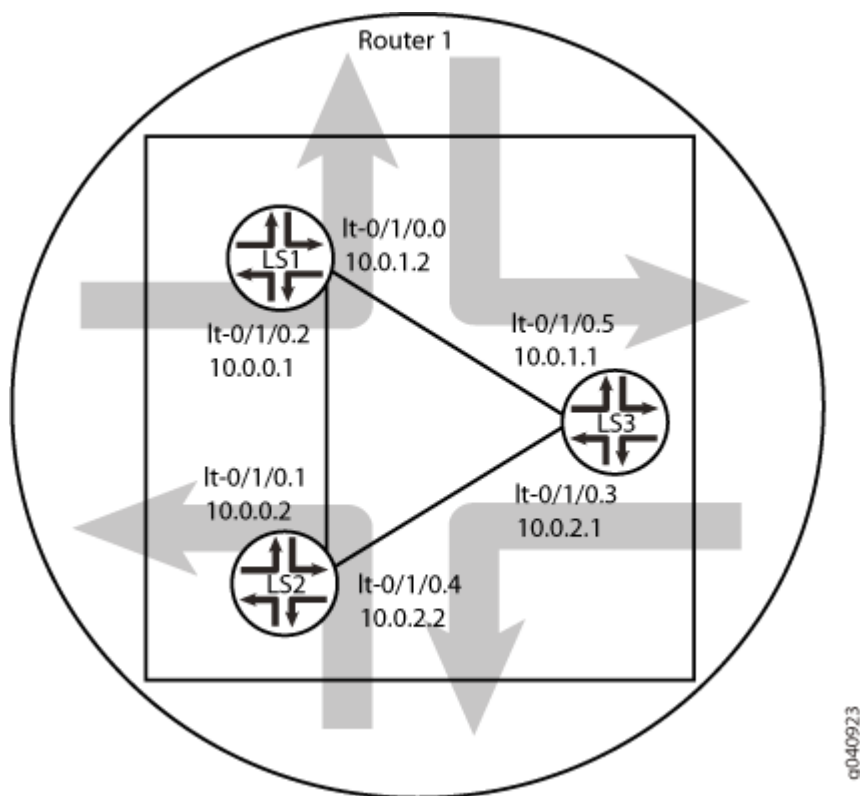
This example shows an IS-IS configuration with three logical systems running on one physical router. Each logical system has its own routing table. The configuration enables the protocol on all logical tunnel interfaces that participate in the IS-IS domain.

## Topology

[Figure 50 on page 723](#) shows the sample network.



Figure 50: IS-IS on Logical Systems



## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 724](#)
- [Procedure | 725](#)
- [Results | 728](#)

## CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter commit from configuration mode.

```

set logical-systems LS1 interfaces lt-0/1/0 unit 2 description LS1->LS2
set logical-systems LS1 interfaces lt-0/1/0 unit 2 encapsulation ethernet
set logical-systems LS1 interfaces lt-0/1/0 unit 2 peer-unit 1
set logical-systems LS1 interfaces lt-0/1/0 unit 2 family inet address 10.0.0.1/30
set logical-systems LS1 interfaces lt-0/1/0 unit 2 family iso
set logical-systems LS1 interfaces lt-0/1/0 unit 0 description LS1->LS3
set logical-systems LS1 interfaces lt-0/1/0 unit 0 encapsulation ethernet
set logical-systems LS1 interfaces lt-0/1/0 unit 0 peer-unit 5
set logical-systems LS1 interfaces lt-0/1/0 unit 0 family inet address 10.0.1.2/30
set logical-systems LS1 interfaces lt-0/1/0 unit 0 family iso
set logical-systems LS1 interfaces lo0 unit 1 family iso address 49.0001.1720.1600.1001.00
set logical-systems LS1 protocols isis interface lt-0/1/0.0
set logical-systems LS1 protocols isis interface lt-0/1/0.2
set logical-systems LS1 protocols isis interface lo0.1 passive
set logical-systems LS2 interfaces lt-0/1/0 unit 1 description LS2->LS1
set logical-systems LS2 interfaces lt-0/1/0 unit 1 encapsulation ethernet
set logical-systems LS2 interfaces lt-0/1/0 unit 1 peer-unit 2
set logical-systems LS2 interfaces lt-0/1/0 unit 1 family inet address 10.0.0.2/30
set logical-systems LS2 interfaces lt-0/1/0 unit 1 family iso
set logical-systems LS2 interfaces lt-0/1/0 unit 4 description LS2->LS3
set logical-systems LS2 interfaces lt-0/1/0 unit 4 encapsulation ethernet
set logical-systems LS2 interfaces lt-0/1/0 unit 4 peer-unit 3
set logical-systems LS2 interfaces lt-0/1/0 unit 4 family inet address 10.0.2.2/30
set logical-systems LS2 interfaces lt-0/1/0 unit 4 family iso
set logical-systems LS2 interfaces lo0 unit 2 family iso address 49.0001.1720.1600.2002.00
set logical-systems LS2 protocols isis interface lt-0/1/0.1
set logical-systems LS2 protocols isis interface lt-0/1/0.4
set logical-systems LS2 protocols isis interface lo0.2 passive
set logical-systems LS3 interfaces lt-0/1/0 unit 3 description LS3->LS2
set logical-systems LS3 interfaces lt-0/1/0 unit 3 encapsulation ethernet
set logical-systems LS3 interfaces lt-0/1/0 unit 3 peer-unit 4
set logical-systems LS3 interfaces lt-0/1/0 unit 3 family inet address 10.0.2.1/30
set logical-systems LS3 interfaces lt-0/1/0 unit 3 family iso
set logical-systems LS3 interfaces lt-0/1/0 unit 5 description LS3->LS1
set logical-systems LS3 interfaces lt-0/1/0 unit 5 encapsulation ethernet
set logical-systems LS3 interfaces lt-0/1/0 unit 5 peer-unit 0

```

```

set logical-systems LS3 interfaces lt-0/1/0 unit 5 family inet address 10.0.1.1/30
set logical-systems LS3 interfaces lt-0/1/0 unit 5 family iso
set logical-systems LS3 interfaces lo0 unit 3 family iso address 49.0001.1234.1600.2231.00
set logical-systems LS3 protocols isis interface lt-0/1/0.5
set logical-systems LS3 protocols isis interface lt-0/1/0.3
set logical-systems LS3 protocols isis interface lo0.3 passive

```

## Procedure

### Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure IS-IS on logical systems:

1. Configure the logical tunnel interface on Logical System LS1 connecting to Logical System LS2.

```

[edit logical-systems LS1]
user@host# set interfaces lt-0/1/0 unit 2 description LS1->LS2
user@host# set interfaces lt-0/1/0 unit 2 encapsulation ethernet
user@host# set interfaces lt-0/1/0 unit 2 peer-unit 1
user@host# set interfaces lt-0/1/0 unit 2 family inet address 10.0.0.1/30
user@host# set interfaces lt-0/1/0 unit 2 family iso

```

2. Configure the logical tunnel interface on Logical System LS1 connecting to Logical System LS3.

```

[edit logical-systems LS1]
user@host# set interfaces lt-0/1/0 unit 0 description LS1->LS3
user@host# set interfaces lt-0/1/0 unit 0 encapsulation ethernet
user@host# set interfaces lt-0/1/0 unit 0 peer-unit 5
user@host# set interfaces lt-0/1/0 unit 0 family inet address 10.0.1.2/30
user@host# set interfaces lt-0/1/0 unit 0 family iso

```

3. Configure the logical tunnel interface on Logical System LS2 connecting to Logical System LS1.

```

[edit logical-systems LS2]
user@host# set interfaces lt-0/1/0 unit 1 description LS2->LS1
user@host# set interfaces lt-0/1/0 unit 1 encapsulation ethernet

```

```

user@host# set interfaces lt-0/1/0 unit 1 peer-unit 2
user@host# set interfaces lt-0/1/0 unit 1 family inet address 10.0.0.2/30
user@host# set interfaces lt-0/1/0 unit 1 family iso

```

4. Configure the logical tunnel interface on Logical System LS2 connecting to Logical System LS3.

```

[edit logical-systems LS2]
user@host# set interfaces lt-0/1/0 unit 4 description LS2->LS3
user@host# set interfaces lt-0/1/0 unit 4 encapsulation ethernet
user@host# set interfaces lt-0/1/0 unit 4 peer-unit 3
user@host# set interfaces lt-0/1/0 unit 4 family inet address 10.0.2.2/30
user@host# set interfaces lt-0/1/0 unit 4 family iso

```

5. Configure the logical tunnel interface on Logical System LS3 connecting to Logical System LS2.

```

[edit logical-systems LS3]
user@host# set interfaces lt-0/1/0 unit 3 description LS3->LS2
user@host# set interfaces lt-0/1/0 unit 3 encapsulation ethernet
user@host# set interfaces lt-0/1/0 unit 3 peer-unit 4
user@host# set interfaces lt-0/1/0 unit 3 family inet address 10.0.2.1/30
user@host# set interfaces lt-0/1/0 unit 3 family iso

```

6. Configure the logical tunnel interface on Logical System LS3 connecting to Logical System LS1.

```

[edit logical-systems LS3]
user@host# set interfaces lt-0/1/0 unit 5 description LS3->LS1
user@host# set interfaces lt-0/1/0 unit 5 encapsulation ethernet
user@host# set interfaces lt-0/1/0 unit 5 peer-unit 0
user@host# set interfaces lt-0/1/0 unit 5 family inet address 10.0.1.1/30
user@host# set interfaces lt-0/1/0 unit 5 family iso

```

7. Configure the ISO address on the loopback interface for the three logical systems.

```
[edit logical-systems LS1]
user@host# set interfaces lo0 unit 1 family iso address 49.0001.1720.1600.1001.00
user@host# set protocols isis interface lo0.1 passive
```

```
[edit logical-systems LS2]
user@host# set interfaces lo0 unit 2 family iso address 49.0001.1720.1600.2002.00
user@host# set protocols isis interface lo0.2 passive
```

```
[edit logical-systems LS3]
user@host# set interfaces lo0 unit 3 family iso address 49.0001.1234.1600.2231.00
user@host# set protocols isis interface lo0.3 passive
```

8. Configure IS-IS on all the interfaces.

```
[edit logical-systems LS1 protocols isis]
user@host# set interface lt-0/1/0.0
user@host# set interface lt-0/1/0.2
```

```
[edit logical-systems LS2 protocols isis]
user@host# set interface lt-0/1/0.1
user@host# set interface lt-0/1/0.4
```

```
[edit logical-systems LS3 protocols isis]
user@host# set interface lt-0/1/0.5
user@host# set interface lt-0/1/0.3
```

9. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

## Results

From configuration mode, confirm your configuration by issuing the `show logical-systems` command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show logical-systems
LS1 {
  interfaces {
    lt-0/1/0 {
      unit 0 {
        description LS1->LS3;
        encapsulation ethernet;
        peer-unit 5;
        family inet {
          address 10.0.1.2/30;
        }
        family iso;
      }
      unit 2 {
        description LS1->LS2;
        encapsulation ethernet;
        peer-unit 1;
        family inet {
          address 10.0.0.1/30;
        }
        family iso;
      }
    }
    lo0 {
      unit 1 {
        family iso {
          address 49.0001.1720.1600.1001.00;
        }
      }
    }
  }
  protocols {
    isis {
      interface lt-0/1/0.0;
      interface lt-0/1/0.2;
      interface lo0.1 {
```

```

        passive;
    }
}
}
LS2 {
    interfaces {
        lt-0/1/0 {
            unit 1 {
                description LS2->LS1;
                encapsulation ethernet;
                peer-unit 2;
                family inet {
                    address 10.0.0.2/30;
                }
                family iso;
            }
            unit 4 {
                description LS2->LS3;
                encapsulation ethernet;
                peer-unit 3;
                family inet {
                    address 10.0.2.2/30;
                }
                family iso;
            }
        }
        lo0 {
            unit 2 {
                family iso {
                    address 49.0001.1720.1600.2002.00;
                }
            }
        }
    }
    protocols {
        isis {
            interface lt-0/1/0.1;
            interface lt-0/1/0.4;
            interface lo0.2 {
                passive;
            }
        }
    }
}

```

```

    }
  }
}
LS3 {
  interfaces {
    lt-0/1/0 {
      unit 3 {
        description LS3->LS2;
        encapsulation ethernet;
        peer-unit 4;
        family inet {
          address 10.0.2.1/30;
        }
        family iso;
      }
      unit 5 {
        description LS3->LS1;
        encapsulation ethernet;
        peer-unit 0;
        family inet {
          address 10.0.1.1/30;
        }
        family iso;
      }
    }
    lo0 {
      unit 3 {
        family iso {
          address 49.0001.1234.1600.2231.00;
        }
      }
    }
  }
  protocols {
    isis {
      interface lt-0/1/0.3;
      interface lt-0/1/0.5;
      interface lo0.3 {
        passive;
      }
    }
  }
}

```



```
    }  
}
```

Verification

IN THIS SECTION

- [Verifying That the Logical Systems Are Up | 731](#)
- [Verifying Connectivity Between the Logical Systems | 732](#)

Confirm that the configuration is working properly.

Verifying That the Logical Systems Are Up

Purpose

Make sure that the interfaces are properly configured.

Action

```
user@host> show interfaces terse
```

Interface	Admin	Link	Proto	Local	Remote
...					
lt-0/1/0	up	up			
lt-0/1/0.0	up	up	inet	10.0.1.2/30	
			iso		
lt-0/1/0.1	up	up	inet	10.0.0.2/30	
			iso		
lt-0/1/0.2	up	up	inet	10.0.0.1/30	
			iso		
lt-0/1/0.3	up	up	inet	10.0.2.1/30	
			iso		
lt-0/1/0.4	up	up	inet	10.0.2.2/30	
			iso		
lt-0/1/0.5	up	up	inet	10.0.1.1/30	
			iso		
...					

## Verifying Connectivity Between the Logical Systems

### Purpose

Make sure that the IS-IS adjacencies are established by checking the logical system routing entries and by pinging the logical systems.

### Action

```
user@host> show route logical-system LS1
inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/30      *[Direct/0] 3w0d 01:37:52
                 > via lt-0/1/0.2
10.0.0.1/32      *[Local/0] 3w0d 01:37:52
                 Local via lt-0/1/0.2
10.0.1.0/30      *[Direct/0] 3w0d 01:37:52
                 > via lt-0/1/0.0
10.0.1.2/32      *[Local/0] 3w0d 01:37:52
                 Local via lt-0/1/0.0
10.0.2.0/30      *[IS-IS/15] 3w0d 01:37:13, metric 20
                 > to 10.0.1.1 via lt-0/1/0.0
                 to 10.0.0.2 via lt-0/1/0.2

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

49.0001.1720.1600.1001/72
                 *[Direct/0] 3w0d 01:37:52
                 > via lo0.1
```

```
user@host> show route logical-system LS2
inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/30      *[Direct/0] 3w0d 01:38:01
                 > via lt-0/1/0.1
10.0.0.2/32      *[Local/0] 3w0d 01:38:01
```

```

                Local via lt-0/1/0.1
10.0.1.0/30      *[IS-IS/15] 3w0d 01:37:01, metric 20
                  to 10.0.0.1 via lt-0/1/0.1
                  > to 10.0.2.1 via lt-0/1/0.4
10.0.2.0/30      *[Direct/0] 3w0d 01:38:01
                  > via lt-0/1/0.4
10.0.2.2/32      *[Local/0] 3w0d 01:38:01
                  Local via lt-0/1/0.4

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

49.0001.1720.1600.2002/72
                *[Direct/0] 3w0d 01:38:01
                > via lo0.2

```

```

user@host> show route logical-system LS3
inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/30      *[IS-IS/15] 3w0d 01:37:10, metric 20
                  to 10.0.2.2 via lt-0/1/0.3
                  > to 10.0.1.2 via lt-0/1/0.5
10.0.1.0/30      *[Direct/0] 3w0d 01:38:10
                  > via lt-0/1/0.5
10.0.1.1/32      *[Local/0] 3w0d 01:38:11
                  Local via lt-0/1/0.5
10.0.2.0/30      *[Direct/0] 3w0d 01:38:11
                  > via lt-0/1/0.3
10.0.2.1/32      *[Local/0] 3w0d 01:38:11
                  Local via lt-0/1/0.3

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

49.0001.1234.1600.2231/72
                *[Direct/0] 3w0d 01:38:11
                > via lo0.3

```

### From LS1, Ping LS3

```
user@host> set cli logical-system LS1
```

```
user@host:LS1> ping 10.0.2.1
PING 10.0.2.1 (10.0.2.1): 56 data bytes
64 bytes from 10.0.2.1: icmp_seq=0 ttl=63 time=1.264 ms
64 bytes from 10.0.2.1: icmp_seq=1 ttl=63 time=1.189 ms
64 bytes from 10.0.2.1: icmp_seq=2 ttl=63 time=1.165 ms
^C
--- 10.0.2.1 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.165/1.206/1.264/0.042 ms
```

### From LS3, Ping LS1

```
user@host> set cli logical-system LS3
```

```
user@host:LS3> ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1): 56 data bytes
64 bytes from 10.0.0.1: icmp_seq=0 ttl=63 time=1.254 ms
64 bytes from 10.0.0.1: icmp_seq=1 ttl=63 time=1.210 ms
^C
--- 10.0.0.1 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.210/1.232/1.254/0.022 ms
```

### From LS1, Ping LS2

```
user@host> set cli logical-system LS1
```

```
user@host:LS1> ping 10.0.2.2
PING 10.0.2.2 (10.0.2.2): 56 data bytes
64 bytes from 10.0.2.2: icmp_seq=0 ttl=64 time=1.240 ms
64 bytes from 10.0.2.2: icmp_seq=1 ttl=64 time=1.204 ms
64 bytes from 10.0.2.2: icmp_seq=2 ttl=64 time=1.217 ms
^C
```

```

--- 10.0.2.2 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.204/1.220/1.240/0.015 ms

```

### From LS2, Ping LS1

```
user@host> set cli logical-system LS2
```

```

user@host:LS2> ping 10.0.1.2
PING 10.0.1.2 (10.0.1.2): 56 data bytes
64 bytes from 10.0.1.2: icmp_seq=0 ttl=64 time=1.308 ms
64 bytes from 10.0.1.2: icmp_seq=1 ttl=64 time=1.235 ms
^C
--- 10.0.1.2 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.235/1.272/1.308/0.037 ms

```

### From LS2, Ping LS3

```
user@host> set cli logical-system LS2
```

```

user@host:LS2> ping 10.0.1.1
PING 10.0.1.1 (10.0.1.1): 56 data bytes
64 bytes from 10.0.1.1: icmp_seq=0 ttl=64 time=1.253 ms
64 bytes from 10.0.1.1: icmp_seq=1 ttl=64 time=1.194 ms
64 bytes from 10.0.1.1: icmp_seq=2 ttl=64 time=1.212 ms
64 bytes from 10.0.1.1: icmp_seq=3 ttl=64 time=1.221 ms
64 bytes from 10.0.1.1: icmp_seq=4 ttl=64 time=1.195 ms
^C
--- 10.0.1.1 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.194/1.215/1.253/0.022 ms

```

## From LS3, Ping LS2

```
user@host> set cli logical-system LS3
```

```
user@host:LS3> ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2): 56 data bytes
64 bytes from 10.0.0.2: icmp_seq=0 ttl=64 time=1.240 ms
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=1.217 ms
^C
--- 10.0.0.2 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.217/1.228/1.240/0.012 ms
```

## RELATED DOCUMENTATION

*Example: Creating an Interface on a Logical System*

*Example: Connecting Logical Systems Within the Same Device Using Logical Tunnel Interfaces on MX Series Routers and EX Series Switches*

## Example: Configuring an IS-IS Default Route Policy on Logical Systems

### IN THIS SECTION

- [Requirements | 737](#)
- [Overview | 737](#)
- [Configuration | 738](#)
- [Verification | 742](#)

This example shows logical systems configured on a single physical router and explains how to configure a default route on one logical system.

## Requirements

No special configuration beyond device initialization is required before configuring this example.

## Overview

### IN THIS SECTION

- [Topology](#) | 737

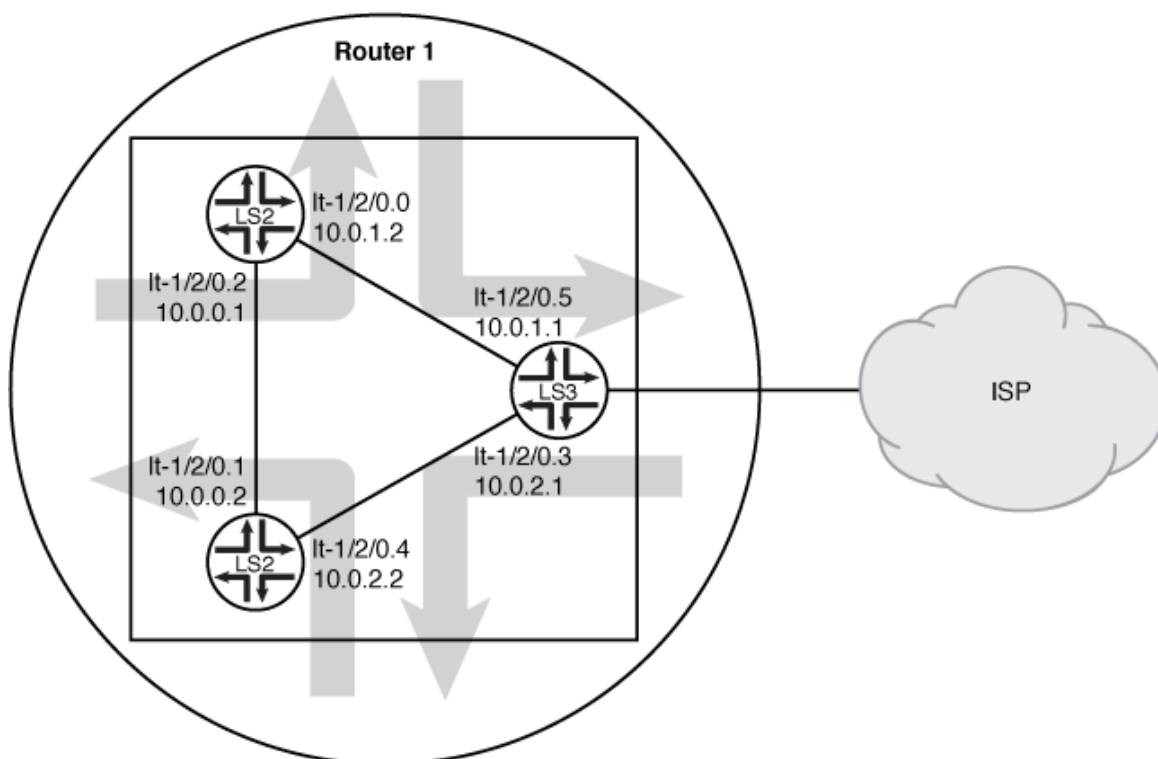
This example shows a logical system redistributing a default route to other logical systems. All logical systems are running IS-IS. A common reason for a default route is to provide a path for sending traffic destined outside the IS-IS domain.

In this example, the default route is not used for forwarding traffic. The `no-install` statement prevents the route from being installed in the forwarding table of Logical System LS3. If you configure a route so it is not installed in the forwarding table, the route is still eligible to be exported from the routing table to other protocols. The `discard` statement silently drops packets without notice.

## Topology

[Figure 51 on page 738](#) shows the sample network.

Figure 51: IS-IS with a Default Route to an ISP



9040918

## Configuration

### IN THIS SECTION

- [CLI Quick Configuration | 738](#)
- [Procedure | 739](#)
- [Results | 741](#)

### CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the [edit] hierarchy level, and then enter commit from configuration mode.

```
set logical-systems LS3 interfaces lt-1/2/0 unit 3 description LS3->LS2
set logical-systems LS3 interfaces lt-1/2/0 unit 3 encapsulation ethernet
```



```

set logical-systems LS3 interfaces lt-1/2/0 unit 3 peer-unit 4
set logical-systems LS3 interfaces lt-1/2/0 unit 3 family inet address 10.0.2.1/30
set logical-systems LS3 interfaces lt-1/2/0 unit 3 family iso
set logical-systems LS3 interfaces lt-1/2/0 unit 5 description LS3->LS1
set logical-systems LS3 interfaces lt-1/2/0 unit 5 encapsulation ethernet
set logical-systems LS3 interfaces lt-1/2/0 unit 5 peer-unit 0
set logical-systems LS3 interfaces lt-1/2/0 unit 5 family inet address 10.0.1.1/30
set logical-systems LS3 interfaces lt-1/2/0 unit 5 family iso
set logical-systems LS3 interfaces lo0 unit 3 family iso address 49.0001.1234.1600.2231.00
set logical-systems LS3 protocols isis export isis-default
set logical-systems LS3 protocols isis interface lt-1/2/0.3
set logical-systems LS3 protocols isis interface lt-1/2/0.5
set logical-systems LS3 protocols isis interface lo0.3 passive
set logical-systems LS3 routing-options static route 0.0.0.0/0 discard
set logical-systems LS3 routing-options static route 0.0.0.0/0 no-install
set logical-systems LS3 policy-options policy-statement isis-default from protocol static
set logical-systems LS3 policy-options policy-statement isis-default from route-filter 0.0.0.0/0
exact
set logical-systems LS3 policy-options policy-statement isis-default then accept

```

## Procedure

### Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [CLI User Guide](#).

To configure an IS-IS default route policy on logical systems:

1. Configure the logical tunnel interfaces.

```

[edit logical-systems LS3 interfaces lt-1/2/0]
user@R1# set unit 3 description LS3->LS2
user@R1# set unit 3 encapsulation ethernet
user@R1# set unit 3 peer-unit 4
user@R1# set unit 3 family inet address 10.0.2.1/30
user@R1# set unit 3 family iso
user@R1# set unit 5 description LS3->LS1
user@R1# set unit 5 encapsulation ethernet
user@R1# set unit 5 peer-unit 0
user@R1# set unit 5 family inet address 10.0.1.1/30

```

```

user@R1# set unit 5 family iso
[edit logical-systems LS3 interfaces lo0 unit 3]
user@R1# set family iso address 49.0001.1234.1600.2231.00

```

2. Enable IS-IS on the interfaces.

```

[edit logical-systems LS3 protocols isis]
user@R1# set interface lt-1/2/0.3
user@R1# set interface lt-1/2/0.5
user@R1# set interface lo0.3 passive

```

3. Configure the default route on Logical System LS3.

```

[edit logical-systems LS3 routing-options]
user@R1# set static route 0.0.0.0/0 discard
user@R1# set static route 0.0.0.0/0 no-install

```

4. Configure the default route policy on Logical System LS3.

```

[edit logical-systems LS3 policy-options]
user@R1# set policy-statement isis-default from protocol static
user@R1# set policy-statement isis-default from route-filter 0.0.0.0/0 exact
user@R1# set policy-statement isis-default then accept

```

5. Apply the export policy to IS-IS on Logical System LS3.

```

[edit logical-systems LS3 protocols isis]
user@R1# set export isis-default

```

6. If you are done configuring the device, commit the configuration.

```

[edit]
user@R1# commit

```

## Results

From configuration mode, confirm your configuration by issuing the `show logical-systems LS3` command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show logical-systems LS3
interfaces {
  lt-1/2/0 {
    unit 3 {
      description LS3->LS2;
      encapsulation ethernet;
      peer-unit 4;
      family inet {
        address 10.0.2.1/30;
      }
      family iso;
    }
    unit 5 {
      description LS3->LS1;
      encapsulation ethernet;
      peer-unit 0;
      family inet {
        address 10.0.1.1/30;
      }
      family iso;
    }
  }
  lo0 {
    unit 3 {
      family iso {
        address 49.0001.1234.1600.2231.00;
      }
    }
  }
}
protocols {
  isis {
    export isis-default;
    interface lt-1/2/0.3;
    interface lt-1/2/0.5;
    interface lo0.3 {
```

```
        passive;
    }
}
}
policy-options {
    policy-statement isis-default {
        from {
            protocol static;
            route-filter 0.0.0.0/0 exact;
        }
        then accept;
    }
}
routing-options {
    static {
        route 0.0.0.0/0 {
            discard;
            no-install;
        }
    }
}
}
```

## Verification

### IN THIS SECTION

- [Verifying That the Static Route Is Redistributed | 742](#)

Confirm that the configuration is working properly.

### Verifying That the Static Route Is Redistributed

#### Purpose

Make sure that the IS-IS policy is working by checking the routing tables.

## Action

```

user@R1> show route logical-system LS3
inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0      *[Static/5] 00:00:45
                Discard
10.0.0.0/30    *[IS-IS/15] 1w0d 10:14:14, metric 20
                to 10.0.2.2 via lt-1/2/0.3
                > to 10.0.1.2 via lt-1/2/0.5
10.0.1.0/30    *[Direct/0] 1w0d 10:15:18
                > via lt-1/2/0.5
10.0.1.1/32    *[Local/0] 1w0d 10:15:18
                Local via lt-1/2/0.5
10.0.2.0/30    *[Direct/0] 1w0d 10:15:18
                > via lt-1/2/0.3
10.0.2.1/32    *[Local/0] 1w0d 10:15:18
                Local via lt-1/2/0.3

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

49.0001.1234.1600.2231/72
                *[Direct/0] 1w0d 10:17:19
                > via lo0.3

```

```

user@R1> show route logical-system LS2
inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0      *[IS-IS/160] 00:01:38, metric 10
                > to 10.0.2.1 via lt-1/2/0.4
10.0.0.0/30    *[Direct/0] 1w0d 10:16:11
                > via lt-1/2/0.1
10.0.0.2/32    *[Local/0] 1w0d 10:16:11
                Local via lt-1/2/0.1
10.0.1.0/30    *[IS-IS/15] 1w0d 10:15:07, metric 20
                > to 10.0.0.1 via lt-1/2/0.1
                to 10.0.2.1 via lt-1/2/0.4
10.0.2.0/30    *[Direct/0] 1w0d 10:16:11

```

```

                > via lt-1/2/0.4
10.0.2.2/32      *[Local/0] 1w0d 10:16:11
                  Local via lt-1/2/0.4

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

49.0001.1720.1600.2002/72
                *[Direct/0] 1w0d 10:18:12
                > via lo0.2

```

```

user@R1> show route logical-system LS1
inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0       *[IS-IS/160] 00:02:01, metric 10
                  > to 10.0.1.1 via lt-1/2/0.0
10.0.0.0/30     *[Direct/0] 1w0d 10:16:34
                  > via lt-1/2/0.2
10.0.0.1/32     *[Local/0] 1w0d 10:16:34
                  Local via lt-1/2/0.2
10.0.1.0/30     *[Direct/0] 1w0d 10:16:34
                  > via lt-1/2/0.0
10.0.1.2/32     *[Local/0] 1w0d 10:16:34
                  Local via lt-1/2/0.0
10.0.2.0/30     *[IS-IS/15] 1w0d 10:15:55, metric 20
                  to 10.0.1.1 via lt-1/2/0.0
                  > to 10.0.0.2 via lt-1/2/0.2

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

49.0001.1720.1600.1001/72
                *[Direct/0] 1w0d 10:18:35
                > via lo0.1

```

## Meaning

The routing table on Logical System LS3 contains the default 0.0.0.0/0 route from protocol Static. The routing tables on Logical System LS1 and Logical System LS2 contain the default 0.0.0.0/0 route from

protocol IS-IS. If Logical System LS1 and Logical System LS2 receive packets destined for networks not specified in their routing tables, those packets will be sent to Logical System LS3 for further processing. This configuration assumes that Logical System LS3 has a connection to an ISP or another external network.

## RELATED DOCUMENTATION

| *Example: Creating an Interface on a Logical System*

# 3

PART

## Monitoring and Troubleshooting Network Issues

---

- Monitoring Networks | **747**
  - Troubleshooting Network Issues | **756**
  - Troubleshooting IS-IS | **766**
-



# Monitoring Networks

## IN THIS CHAPTER

- [Example: Tracing Global Routing Protocol Operations | 747](#)
- [Understanding IS-IS Subscribe Configuration | 754](#)
- [IS-IS Purge Originator Identification Overview | 755](#)

## Example: Tracing Global Routing Protocol Operations

## IN THIS SECTION

- [Requirements | 747](#)
- [Overview | 748](#)
- [Configuration | 748](#)
- [Verification | 753](#)

This example shows how to list and view files that are created when you enable global routing trace operations.

### Requirements

You must have the **view** privilege.

## Overview

To configure global routing protocol tracing, include the `traceoptions` statement at the `[edit routing-options]` hierarchy level:

```
traceoptions {
  file filename <files number> <size size> <world-readable | no-world-readable>;
  flag flag <disable>;
}
```

The flags in a `traceoptions` `flag` statement are identifiers. When you use the `set` command to configure a flag, any flags that might already be set are not modified. In the following example, setting the **timer** tracing flag has no effect on the already configured **task** flag. Use the `delete` command to delete a particular flag.

```
[edit routing-options traceoptions]
user@host# show
flag task;
user@host# set traceoptions flag timer
user@host# show
flag task;
flag timer;
user@host# delete traceoptions flag task
user@host# show
flag timer;
```

This example shows how to configure and view a trace file that tracks changes in the routing table. The steps can be adapted to apply to trace operations for any Junos OS hierarchy level that supports trace operations.



**TIP:** To view a list of hierarchy levels that support tracing operations, enter the `help apropos traceoptions` command in configuration mode.

## Configuration

### IN THIS SECTION

● [CLI Quick Configuration | 749](#)

- [Configuring Trace Operations | 749](#)
- [Viewing the Trace File | 750](#)
- [Results | 753](#)

## CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

```
set routing-options traceoptions file routing-table-changes
set routing-options traceoptions file size 10m
set routing-options traceoptions file files 10
set routing-options traceoptions flag route
set routing-options static route 1.1.1.2/32 next-hop 10.0.45.6
```

## Configuring Trace Operations

### Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the [Junos OS CLI User Guide](#).

To configure the trace operations:

1. Configure trace operations.

```
[edit routing-options traceoptions]
user@host# set file routing-table-changes
user@host# set file size 10m
user@host# set file files 10
user@host# set flag route
```

2. Configure a static route to cause a change in the routing table.

```
[edit routing-options static]
user@host# set route 1.1.1.2/32 next-hop 10.0.45.6
```

3. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

## Viewing the Trace File

### Step-by-Step Procedure

To view the trace file:

1. In operational mode, list the log files on the system.

```
user@host> file list /var/log
/var/log:
...
routing-table-changes
...
```

2. View the contents of the **routing-table-changes** file.

```
user@host> file show /var/log/routing-table-changes
Dec 15 11:09:29 trace_on: Tracing to "/var/log/routing-table-changes" started
Dec 15 11:09:29.496507
Dec 15 11:09:29.496507 Tracing flags enabled: route
Dec 15 11:09:29.496507
Dec 15 11:09:29.533203 inet_routerid_notify: Router ID: 192.168.4.1
Dec 15 11:09:29.533334 inet_routerid_notify: No Router ID assigned
Dec 15 11:09:29.533381 inet_routerid_notify: No Router ID assigned
Dec 15 11:09:29.533420 inet_routerid_notify: No Router ID assigned
Dec 15 11:09:29.534915 inet_routerid_notify: Router ID: 192.168.4.1
Dec 15 11:09:29.542934 inet_routerid_notify: No Router ID assigned
Dec 15 11:09:29.549253 inet_routerid_notify: No Router ID assigned
Dec 15 11:09:29.556878 inet_routerid_notify: No Router ID assigned
```

```
Dec 15 11:09:29.582990 rt_static_reinit: examined 3 static nexthops, 0 unreferenced
Dec 15 11:09:29.589920
Dec 15 11:09:29.589920 task_reconfigure reinitializing done
...
```

### 3. Filter the output of the log file.

```
user@host> file show /var/log/routing-table-changes | match 1.1.1.2
Dec 15 11:15:30.780314 ADD      1.1.1.2/32          nhid 0 gw 10.0.45.6      Static  pref
5/0 metric  at-0/2/0.0 <ctive Int Ext>
Dec 15 11:15:30.782276 KRT Request: send len 216 v104 seq 0 ADD route/user af 2 table 0 infot
0 addr 1.1.1.2 nhop-type unicast nhindex 663
```

### 4. View the tracing operations in real time by running the `monitor start` command with an optional `match` condition.

```
user@host> monitor start routing-table-changes | match 1.1.1.2
Aug 10 19:21:40.773467 BGP RECV      0.0.0.0/0
Aug 10 19:21:40.773685 bgp_rcv_nlri: 0.0.0.0/0
Aug 10 19:21:40.773778 bgp_rcv_nlri: 0.0.0.0/0 belongs to meshgroup
Aug 10 19:21:40.773832 bgp_rcv_nlri: 0.0.0.0/0 qualified bnp->ribact 0x0 l2afcb 0x0
```

### 5. Deactivate the static route.

```
user@host# deactivate routing-options static route 1.1.1.2/32
user@host# commit
```

```
*** routing-table-changes ***
Dec 15 11:42:59.355557 CHANGE  1.1.1.2/32          nhid 663 gw 10.0.45.6      Static  pref
5/0 metric  at-0/2/0.0 <Delete Int Ext>
Dec 15 11:42:59.426887 KRT Request: send len 216 v104 seq 0 DELETE route/user af 2 table 0
infot 0 addr 1.1.1.2 nhop-type discard filtidx 0
Dec 15 11:42:59.427366 RELEASE 1.1.1.2/32          nhid 663 gw 10.0.45.6      Static  pref
5/0 metric  at-0/2/0.0 <Release Delete Int Ext>
```

6. Halt the `monitor` command by pressing Enter and typing **monitor stop**.

```
[Enter]  
user@host> monitor stop
```

7. When you are finished troubleshooting, consider deactivating trace logging to avoid any unnecessary impact to system resources.

When configuration is deactivated, it appears in the configuration with the **inactive** tag.

```
[edit routing-options]  
user@host# deactivate traceoptions  
user@host# commit
```

```
[edit routing-options]  
user@host# show  
  
inactive: traceoptions {  
  file routing-table-changes size 10m files 10;  
  flag route;  
}  
static {  
  inactive: route 1.1.1.2/32 next-hop 10.0.45.6;  
}
```

8. To reactivate trace operations, use the **activate** configuration-mode statement.

```
[edit routing-options]  
user@host# activate traceoptions  
user@host# commit
```

## Results

From configuration mode, confirm your configuration by entering the `show routing-options` command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show routing-options
traceoptions {
  file routing-table-changes size 10m files 10;
  flag route;
}
static {
  route 1.1.1.2/32 next-hop 10.0.45.6;
}
```

## Verification

### IN THIS SECTION

- [Verifying That the Trace Log File Is Operating | 753](#)

Confirm that the configuration is working properly.

### Verifying That the Trace Log File Is Operating

#### Purpose

Make sure that events are being written to the log file.

#### Action

```
user@host> show log routing-table-changes
Dec 15 11:09:29 trace_on: Tracing to "/var/log/routing-table-changes" started
```

## Understanding IS-IS Subscribe Configuration

### Overview

You can access the segment-routing or spring sensor-based traffic statistics stored in the Packet Forwarding Engine using the sensor ID on the FPC command line. Accessing each FPC to fetch the traffic statistics involves multiple task.

You can now easily record the recent spring sensor-based traffic statistics at the [edit protocols isis source-packet-routing] hierarchy level. Use the following set configuration statements to trigger the on-box spring sensor subscription process and record detailed sensor-based traffic statistics.

```
[edit protocols isis source-packet-routing]
user@host#set sensor-based-stats per-interface-per-member-link <ingress> | <egress>
[edit protocols isis source-packet-routing]
user@host#set sensor-based-stats per-sid <ingress> | <egress>
[edit protocols isis source-packet-routing]
user@host#set sensor-based-stats subscribe interval interval
```

The spring sensor-based subscribe configuration creates a connection with network-agent and subscribes to the configured sensor path. The sensor path is installed in the Packet Forwarding Engine where the telemetry sensor-based data is stored. The telemetry process keeps track of the latest updates received on each subscribed sensor. The show spring traffic statistics command displays the traffic statistics information streamed from the Packet Forwarding Engine.

### Benefits

- Easy access to spring sensor-based traffic statistics is possible on enabling subscribe interval *interval* configuration.

### RELATED DOCUMENTATION

[sensor-based-stats](#)

[show spring interface traffic-statistics](#)



## IS-IS Purge Originator Identification Overview

When the IS-IS protocol purges entries from IS-IS link-state database, there is no way to identify the origin of the purge. If there is a need to investigate the cause of the purge, it is difficult to determine the Intermediate system (IS) that initiated the purge. RFC 6232, *Purge Originator Identification TLV for IS-IS* defines a type, length, and value (TLV) that can be added to the purges, to record the system ID of the IS that had initiated the purge. If an IS generates a purge, this TLV is included in the purge, which also has the system ID of the IS. If an IS receives a purge, the Link State Protocol Data Unit (LSP) flooding does not change the LSP contents, and the TLV is propagated with the purge itself. If an IS receives a purge that does not include this TLV, it adds this TLV with both its own system ID and the system ID of the IS from which it received the purge. This allows the IS that receives this purge to log the system ID of the originator, or the upstream source of the purge. This makes it easier to locate the origin of the purge and its cause. This TLV is also helpful in lab environments.

There is a possibility that during a network attack, a low lifetime is generated maliciously for an LSP, which can initiate a purge on timeout. These LSPs with low lifetime need to be filtered out to avoid purges triggered by a low lifetime LSP.

# Troubleshooting Network Issues

## IN THIS CHAPTER

- [Working with Problems on Your Network | 756](#)
- [Isolating a Broken Network Connection | 757](#)
- [Identifying the Symptoms of a Broken Network Connection | 759](#)
- [Isolating the Causes of a Network Problem | 761](#)
- [Taking Appropriate Action for Resolving the Network Problem | 762](#)
- [Evaluating the Solution to Check Whether the Network Problem Is Resolved | 764](#)

## Working with Problems on Your Network

### IN THIS SECTION

- [Problem | 756](#)
- [Solution | 757](#)

### Problem

#### Description

This checklist provides links to troubleshooting basics, an example network, and includes a summary of the commands you might use to diagnose problems with the router and network.

## Solution

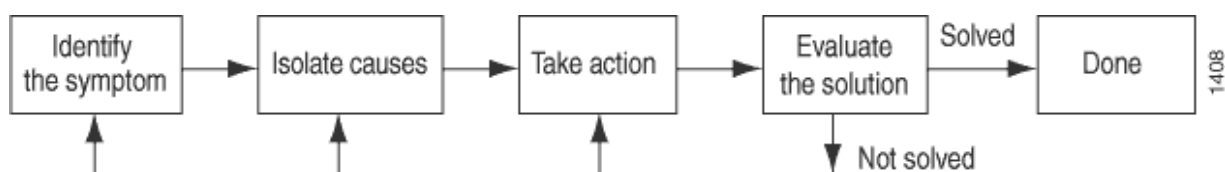
Table 9: Checklist for Working with Problems on Your Network

Tasks	Command or Action
<i>Isolating a Broken Network Connection</i>	
1. <i>Identifying the Symptoms of a Broken Network Connection</i>	<b>ping (ip-address   hostname) show route (ip-address   hostname) traceroute (ip-address   hostname)</b>
1. <i>Isolating the Causes of a Network Problem</i>	show < configuration   interfaces   protocols   route >
1. <i>Taking Appropriate Action for Resolving the Network Problem</i>	[edit] delete routing options static route destination-prefix commit and-quit show route destination-prefix
1. <i>Evaluating the Solution to Check Whether the Network Problem Is Resolved</i>	show route (ip-address   hostname) ping (ip-address   hostname) count 3 traceroute (ip-address   hostname)

## Isolating a Broken Network Connection

By applying the standard four-step process illustrated in [Figure 52 on page 757](#), you can isolate a failed node in the network. Note that the functionality described in this section is not supported in versions 15.1X49, 15.1X49-D30, or 15.1X49-D40.

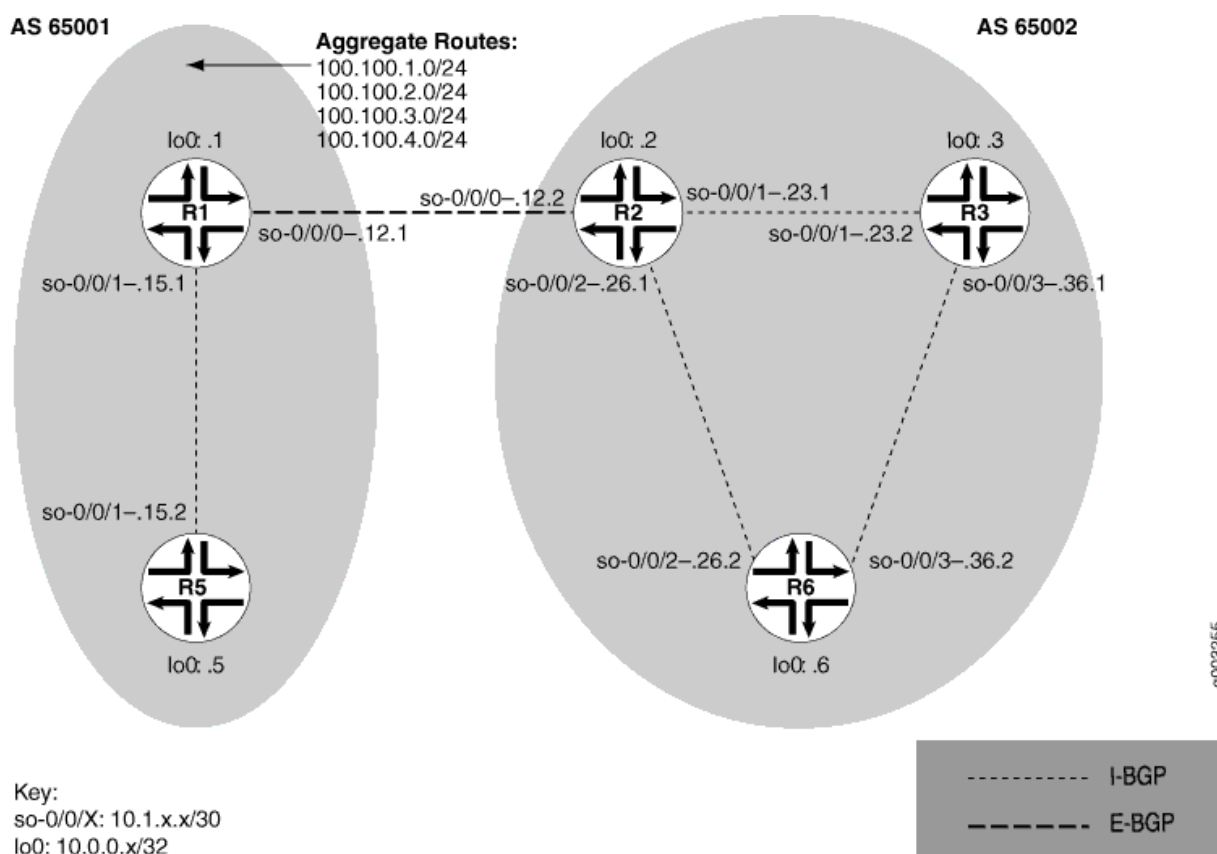
Figure 52: Process for Diagnosing Problems in Your Network



Before you embark on the four-step process, however, it is important that you are prepared for the inevitable problems that occur on all networks. While you might find a solution to a problem by simply trying a variety of actions, you can reach an appropriate solution more quickly if you are systematic in your approach to the maintenance and monitoring of your network. To prepare for problems on your network, understand how the network functions under normal conditions, have records of baseline network activity, and carefully observe the behavior of your network during a problem situation.

Figure 53 on page 758 shows the network topology used in this topic to illustrate the process of diagnosing problems in a network.

Figure 53: Network with a Problem



The network in Figure 53 on page 758 consists of two autonomous systems (ASs). AS 65001 includes two routers, and AS 65002 includes three routers. The border router (R1) in AS 65001 announces aggregated prefixes `100.100.0/24` to the AS 65002 network. The problem in this network is that R6 does not have access to R5 because of a loop between R2 and R6.

To isolate a failed connection in your network, follow the steps in these topics:

- *Isolating the Causes of a Network Problem*

- *Taking Appropriate Action for Resolving the Network Problem*
- *Taking Appropriate Action for Resolving the Network Problem*
- *Evaluating the Solution to Check Whether the Network Problem Is Resolved*

## Identifying the Symptoms of a Broken Network Connection

### IN THIS SECTION

- Problem | 759
- Solution | 759

### Problem

### Description

The symptoms of a problem in your network are usually quite obvious, such as the failure to reach a remote host.

### Solution

To identify the symptoms of a problem on your network, start at one end of your network and follow the routes to the other end, entering all or one of the following Junos OS command-line interfaces (CLI) operational mode commands:

```
user@host> ping (ip-address | host-name)
user@host> show route (ip-address | host-name)
user@host> traceroute (ip-address | host-name)
```

### Sample Output

```
user@R6> ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
```

```

 4  5  00 0054 e2db  0 0000 01 01 a8c6 10.1.26.2 10.0.0.5

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS  Len  ID Flg  off TTL Pro  cks      Src      Dst
 4  5  00 0054 e2de  0 0000 01 01 a8c3 10.1.26.2 10.0.0.5

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS  Len  ID Flg  off TTL Pro  cks      Src      Dst
 4  5  00 0054 e2e2  0 0000 01 01 a8bf 10.1.26.2 10.0.0.5

^C
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 0 packets received, 100% packet loss

user@R6> show route 10.0.0.5

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32          *[IS-IS/165] 00:02:39, metric 10
                    > to 10.1.26.1 via so-0/0/2.0

user@R6> traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 40 byte packets
 1  10.1.26.1 (10.1.26.1)  0.649 ms  0.521 ms  0.490 ms
 2  10.1.26.2 (10.1.26.2)  0.521 ms  0.537 ms  0.507 ms
 3  10.1.26.1 (10.1.26.1)  0.523 ms  0.536 ms  0.514 ms
 4  10.1.26.2 (10.1.26.2)  0.528 ms  0.551 ms  0.523 ms
 5  10.1.26.1 (10.1.26.1)  0.531 ms  0.550 ms  0.524 ms

```

## Meaning

The sample output shows an unsuccessful ping command in which the packets are being rejected because the time to live is exceeded. The output for the `show route` command shows the interface (10.1.26.1) that you can examine further for possible problems. The `traceroute` command shows the loop between 10.1.26.1 (R2) and 10.1.26.2 (R6), as indicated by the continuous repetition of the two interface addresses.

# Isolating the Causes of a Network Problem

IN THIS SECTION

- Problem | 761
- Solution | 761

## Problem

### Description

A particular symptom can be the result of one or more causes. Narrow down the focus of your search to find each individual cause of the unwanted behavior.

### Solution

To isolate the cause of a particular problem, enter one or all of the following Junos OS CLI operational mode command:

```
user@host> show < configuration | bgp | interfaces | isis | ospf | route
>
```

Your particular problem may require the use of more than just the commands listed above. See the appropriate command reference for a more exhaustive list of commonly used operational mode commands.

### Sample Output

```
user@R6> show interfaces terse
Interface           Admin Link Proto Local           Remote
so-0/0/0            up   up
so-0/0/0.0          up   up   inet 10.1.56.2/30
                   iso
so-0/0/2            up   up
so-0/0/2.0          up   up   inet 10.1.26.2/30
                   iso
so-0/0/3            up   up
```

```
so-0/0/3.0          up    up    inet  10.1.36.2/30
                               iso
[...Output truncated...]
```

The following sample output is from R2:

```
user@R2> show route 10.0.0.5

inet.0: 22 destinations, 25 routes (22 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32          *[Static/5] 00:16:21
                    > to 10.1.26.2 via so-0/0/2.0
                    [BGP/170] 3d 20:23:35, MED 5, localpref 100
                    AS path: 65001 I
                    > to 10.1.12.1 via so-0/0/0.0
```

### Meaning

The sample output shows that all interfaces on R6 are up. The output from R2 shows that a static route [Static/5] configured on R2 points to R6 (10.1.26.2) and is the preferred route to R5 because of its low preference value. However, the route is looping from R2 to R6, as indicated by the missing reference to R5 (10.1.15.2).

## Taking Appropriate Action for Resolving the Network Problem

### IN THIS SECTION

- Problem | 763
- Solution | 763



## Problem

### Description

The appropriate action depends on the type of problem you have isolated. In this example, a static route configured on R2 is deleted from the [routing-options] hierarchy level. Other appropriate actions might include the following:

### Solution

- Check the local router's configuration and edit it if appropriate.
- Troubleshoot the intermediate router.
- Check the remote host configuration and edit it if appropriate.
- Troubleshoot routing protocols.
- Identify additional possible causes.

To resolve the problem in this example, enter the following Junos OS CLI commands:

```
[edit]
user@R2# delete routing-options static route destination-
prefix
user@R2# commit and-quit
user@R2# show route destination-prefix
```

### Sample Output

```
[edit]
user@R2# delete routing-options static route 10.0.0.5/32

[edit]
user@R2# commit and-quit
commit complete
Exiting configuration mode

user@R2> show route 10.0.0.5

inet.0: 22 destinations, 24 routes (22 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
10.0.0.5/32      *[BGP/170] 3d 20:26:17, MED 5, localpref 100
                  AS path: 65001 I
                  > to 10.1.12.1 via so-0/0/0.0
```

### Meaning

The sample output shows the static route deleted from the [routing-options] hierarchy and the new configuration committed. The output for the `show route` command now shows the BGP route as the preferred route, as indicated by the asterisk (\*).

## Evaluating the Solution to Check Whether the Network Problem Is Resolved

### IN THIS SECTION

● Problem | 764

● Solution | 765

### Problem

#### Description

If the problem is solved, you are finished. If the problem remains or a new problem is identified, start the process over again.

You can address possible causes in any order. In relation to the network in *Isolating a Broken Network Connection*, we chose to work from the local router toward the remote router, but you might start at a different point, particularly if you have reason to believe that the problem is related to a known issue, such as a recent change in configuration.

## Solution

To evaluate the solution, enter the following Junos OS CLI commands:

```

user@host> show route (ip-address | host-name)
user@host> ping (ip-address | host-name)
user@host> traceroute (ip-address | host-name)

```

## Sample Output

```

user@R6> show route 10.0.0.5

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32          *[BGP/170]  00:01:35, MED 5, localpref 100, from 10.0.0.2
                    AS path: 65001 I
                    > to 10.1.26.1 via so-0/0/2.0

user@R6> ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
64 bytes from 10.0.0.5: icmp_seq=0 ttl=253 time=0.866 ms
64 bytes from 10.0.0.5: icmp_seq=1 ttl=253 time=0.837 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=253 time=0.796 ms
^C
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.796/0.833/0.866/0.029 ms

user@R6> traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 40 byte packets
 1  10.1.26.1 (10.1.26.1)  0.629 ms  0.538 ms  0.497 ms
 2  10.1.12.1 (10.1.12.1)  0.534 ms  0.538 ms  0.510 ms
 3  10.0.0.5 (10.0.0.5)  0.776 ms  0.705 ms  0.672 ms

```

## Meaning

The sample output shows that there is now a connection between R6 and R5. The `show route` command shows that the BGP route to R5 is preferred, as indicated by the asterisk (\*). The `ping` command is successful and the `traceroute` command shows that the path from R6 to R5 is through R2 (10.1.26.1), and then through R1 (10.1.12.1).

# Troubleshooting IS-IS

## IN THIS CHAPTER

- Verifying the IS-IS Protocol | 766
- Verifying the IS-IS Configuration on a Router in a Network | 780
- Displaying the Status of IS-IS Adjacencies | 790
- Displaying Detailed IS-IS Protocol Information | 795
- Analyzing IS-IS Link-State PDUs in Detail | 799
- Displaying Sent or Received IS-IS Protocol Packets | 802

## Verifying the IS-IS Protocol

### IN THIS SECTION

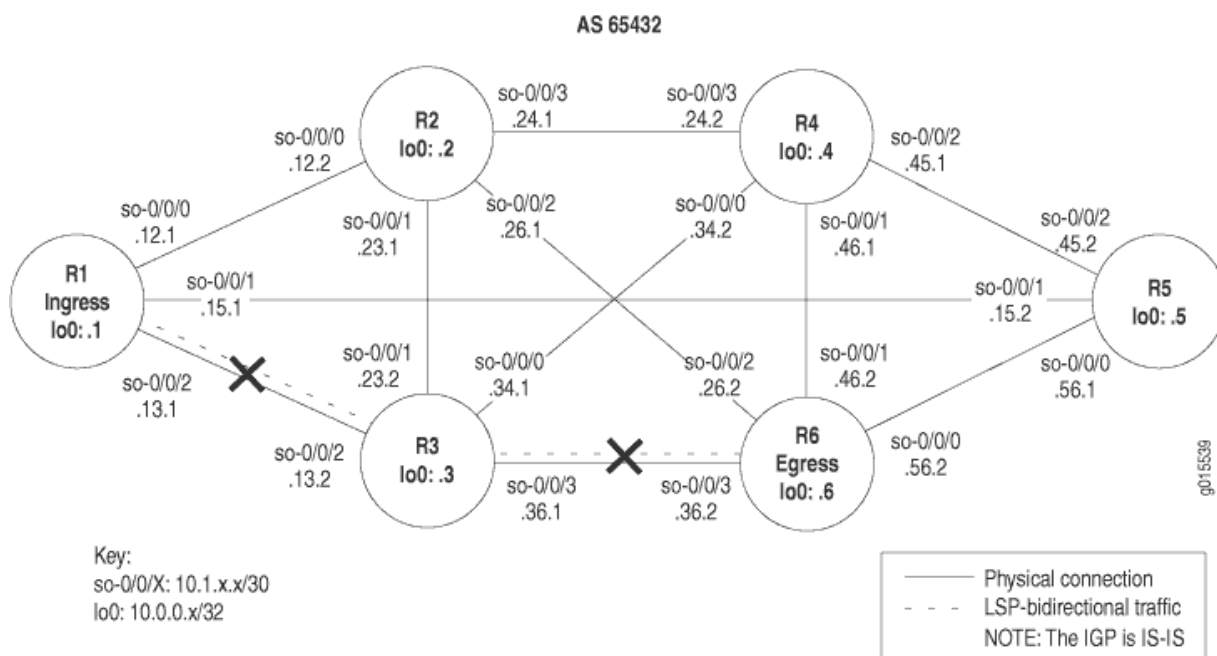
- Verify the LSP | 767
- Verify IS-IS Adjacencies and Interfaces | 770
- Verify the IS-IS Configuration | 772
- Take Appropriate Action | 774
- Verify the LSP Again | 776

### Purpose

If your MPLS network is configured with IS-IS as the interior gateway protocol (IGP), and the output of the `show mpls lsp extensive` command shows that there is a problem, check the IP and IS-IS layers. Because IS-IS and IP are independent of each other, you can check either layer first. For more information about checking the IP layer, see *Verifying the IP Layer*.

After you have checked the IP layer and determined that there is still a problem, check the IS-IS layer, verify that IS-IS adjacencies are up, and make sure that the interfaces and IS-IS protocol are configured correctly.

Figure 54: MPLS Network Broken at the IS-IS Protocol Layer



To check the IS-IS protocol, follow these steps:

## Verify the LSP

### IN THIS SECTION

- Purpose | 767
- Action | 768
- Meaning | 770

### Purpose

Confirm that interfaces are configured for IS-IS, that the IS-IS protocol is configured correctly, and that adjacencies are established.

## Action

To verify the label-switched path (LSP), enter the following command on the ingress, transit, and egress routers:

```
user@host> show mpls lsp extensive
```

## Sample Output 1

### command-name

```
user@R1> show mpls lsp extensive
Ingress LSP: 1 sessions

10.0.0.6
  From: 10.0.0.1, State: Dn,  ActiveRoute: 0 ,  LSPname: R1-to-R6
  ActivePath: (none)
  LoadBalance: Random
  Encoding type: Packet, Switching type: Packet, GPID: IPv4
  Primary                               State: Dn
    24 Oct 21 13:48:01  No Route toward dest [3 times]
    23 Oct 21 13:47:44  Deselected as active
    22 Oct 21 13:47:43  No Route toward dest[2 times]
    21 Oct 21 13:47:43  ResvTear received
    20 Oct 21 13:47:43  Down
    19 Oct 21 13:47:43  10.1.13.2: No Route toward dest[2 times]
    18 Oct 21 13:47:38  Record Route:  10.1.13.2 10.1.36.2
    [...Output truncated...]
  Created: Tue Oct 19 21:22:53 2004
Total 1 displayed, Up 0,  Down 1

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

## Sample Output 2

### command-name

```
user@R3> show mpls lsp extensive
Ingress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

## Sample Output 3

### command-name

```
user@R6> show mpls lsp extensive
Ingress LSP: 1 sessions

10.0.0.1
  From: 10.0.0.6, State: Dn, ActiveRoute: 0 , LSPname: R6-to-R1
  ActivePath: (none)
  LoadBalance: Random
  Encoding type: Packet, Switching type: Packet, GPID: IPv4
  Primary                               State: Dn
    Will be enqueued for recomputation in 3 second(s).
    13 Oct 21 14:23:33 CSPF failed: no route toward 10.0.0.1[90 times]
    12 Oct 21 13:39:56 Deselected as active
    11 Oct 21 13:39:56 CSPF: could not determine self
    [...Output truncated...]
  Created: Tue Oct 19 22:28:30 2004
Total 1 displayed, Up 0, Down 1

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

```
Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
```

Meaning

The sample output shows that LSP R1-to-R6 and the reverse LSP R6-to-R1 are down, and there are no LSP sessions on transit router R3.

Verify IS-IS Adjacencies and Interfaces

IN THIS SECTION

Purpose | 770

Action | 770

Meaning | 772

Purpose

When you check the IS-IS layer, you verify that IS-IS adjacencies are up and that the IS-IS interfaces are included at the protocol level.

Action

To verify the functioning of adjacent interfaces, enter the following commands from the relevant routers:

```
user@host> show isis adjacency
user@host> show isis interface
```

Sample Output 1

command-name

```
user@R1> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/0.0     R2          2 Up         20
```



```

so-0/0/1.0      R5      2 Up      23
so-0/0/2.0      R3      2 Up      26

user@R3> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/0.0      R4      2 Up      23
so-0/0/1.0      R2      2 Up      21
so-0/0/2.0      R1      2 Up      19
so-0/0/3.0      R6      2 Down      0

user@R6> show isis adjacency
IS-IS instance is not running

```

## Sample Output 2

### command-name

```

user@R1> show isis interface
IS-IS interface database:
Interface      L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0          0  0x1 Passive           Passive          0/0
so-0/0/0.0     2  0x1 Disabled          Point to Point   10/10
so-0/0/1.0     2  0x1 Disabled          Point to Point   10/10
so-0/0/2.0     2  0x1 Disabled          Point to Point   10/10

user@R3> show isis interface
IS-IS interface database:
Interface      L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0          0  0x1 Passive           Passive          0/0
so-0/0/0.0     2  0x1 Disabled          Point to Point   10/10
so-0/0/1.0     2  0x1 Disabled          Point to Point   10/10
so-0/0/2.0     2  0x1 Disabled          Point to Point   10/10
so-0/0/3.0     2  0x1 Disabled          Point to Point   10/10

user@R6> show isis interface
IS-IS interface database:
Interface      L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0          0  0x1 Passive           Passive          0/0
so-0/0/0.0     1  0x1 Point to Point     Disabled         10/10
so-0/0/1.0     1  0x1 Down              Disabled         10/10

```

so-0/0/2.0	1	0x1 Point to Point	Disabled	10/10
so-0/0/3.0	1	0x1 Point to Point	Disabled	10/10

Meaning

Sample Output 1 shows that ingress router R1 has established adjacencies with the relevant routers. Transit router R3 does not have an adjacency with egress router R6, and egress router R6 has no adjacencies established in the network shown in *MPLS Network Broken at the IP and IGP Layers*, indicating that the problem might be at the IS-IS protocol level.

Sample Output 2 shows that R1 and R2 are Level 2 routers, in contrast to R6 which is a Level 1 router. When a router is configured explicitly as a Level 1 or Level 2 router, it does not communicate with routers configured at a different level. Level 1 routers communicate with other Level 1 routers within their area, while Level 2 routers communicate with other Level 2 routers, and toward other autonomous systems. Because all the routers in this network are configured for Level 2, they cannot form an adjacency with R6, which is incorrectly configured as a Level 1 router.

SEE ALSO

- [Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding | 22](#)
- [Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups | 21](#)

Verify the IS-IS Configuration

IN THIS SECTION

- [Purpose | 772](#)
- [Action | 773](#)
- [Meaning | 774](#)

Purpose

When you have determined that the problem is probably at the IS-IS protocol level, check the IS-IS configuration of the routers in your network.

## Action

To verify the IS-IS configuration, enter the following command from the relevant routers:

```
user@host> show configuration protocols isis
```

## Sample Output

### command-name

```
user@R1> show configuration protocols isis
level 1 disable;
interface so-0/0/0.0;
interface so-0/0/1.0;
interface so-0/0/2.0;
interface lo0.0; {
    passive

user@R3> show configuration protocols isis
level 1 disable;
interface all {
    level 2 metric 10;
}
interface fxp0.0 {
    disable;
}
interface lo0.0; {
    passive

user@R6> show configuration protocols isis
level 2 disable; <<< Incorrect level disabled
interface all {
    level 2 metric 10;
}
interface fxp0.0 {
    disable;
}
interface lo0.0; {
    passive
```

## Meaning

The sample output shows that R6 has Level 2 disabled, while R1 and R3 have Level 1 disabled. For IS-IS adjacencies to establish, routers need to be at the same level. Another common configuration error is to omit the loopback interface (lo0) from the configuration at the `[edit protocols isis]` hierarchy level. IS-IS does not function correctly if the loopback interface (lo0) is not configured at this level. In addition, including the `passive` statement ensures that protocols are not run over the loopback interface (lo0) and that the loopback interface (lo0) is advertised correctly throughout the network.

## Take Appropriate Action

### IN THIS SECTION

● [Problem | 774](#)

● [Solution | 774](#)

## Problem

### Description

Depending on the error you encountered in your investigation, you must take the appropriate action to correct the problem. In the example below, the routers are configured to function at different levels of the IS-IS protocol.

### Solution

To correct the error in this example, enter the following commands:

### Sample Output

```
[edit protocols isis]
user@R6# show
level 2 disable;
interface all {
    level 2 metric 10;
}
interface fxp0.0 {
    disable;
}
```

```

interface lo0.0; {
passive

[edit protocols isis]
user@R6# delete level 2

[edit protocols isis]
user@R6# set level 1 disable

[edit protocols isis]
user@R6# show
level 1 disable;
interface all {
    level 2 metric 10;
}
interface fxp0.0 {
    disable;
}
interface lo0.0; {
passive

[edit protocols isis]
user@R6# commit
commit complete

[edit protocols isis]
user@R6# run show isis adjacency

```

Interface	System	L State	Hold (secs)	SNPA
so-0/0/0.0	R5	2 Up	22	
so-0/0/1.0	R4	2 Up	22	
so-0/0/2.0	R2	2 Up	22	
so-0/0/3.0	R3	2 Up	22	

**Meaning**

The sample output shows that the configuration error on egress router R6 has been corrected, and IS-IS adjacencies are now established.

**SEE ALSO**

| [IS-IS User Guide](#)

## Verify the LSP Again

### IN THIS SECTION

- Purpose | 776
- Action | 776
- Meaning | 780

### Purpose

After taking the appropriate action to correct the error, the label-switched path (LSP) needs to be checked again to confirm that the problem in the RSVP layer has been resolved.

### Action

To verify that the LSP is up and traversing the network as expected, enter the following command from the ingress, egress, and transit routers:

```
user@host> show mpls lsp extensive
```

### Sample Output 1

#### command-name

```
user@R1> show mpls lsp extensive
Ingress LSP: 1 sessions

10.0.0.6
  From: 10.0.0.1, State: Up,  ActiveRoute: 1 ,  LSPname: R1-to-R6
  ActivePath: (primary)
  LoadBalance: Random
  Encoding type: Packet, Switching type: Packet, GPID: IPv4
  *Primary                               State: Up
    Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 20)
  10.1.13.2 S 10.1.36.2 S
    Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt):
      10.1.13.2 10.1.36.2
```

```

5 Oct 21 15:52:07 Selected as active path
4 Oct 21 15:52:07 Record Route: 10.1.13.2 10.1.36.2
3 Oct 21 15:52:07 Up
2 Oct 21 15:52:07 Originate Call
1 Oct 21 15:52:07 CSPF: computation result accepted
Created: Thu Oct 21 15:52:06 2004
Total 1 displayed, Up 1 , Down 0

```

Egress LSP: 1 sessions

```

10.0.0.1
From: 10.0.0.6, LSPstate: Up, ActiveRoute: 0
  LSPname: R6-to-R1 , LSPpath: Primary
Suggested label received: -, Suggested label sent: -
Recovery label received: -, Recovery label sent: -
Resv style: 1 FF, Label in: 3, Label out: -
Time left: 142, Since: Thu Oct 21 15:41:59 2004
Tspec: rate 0bps size 0bps peak Infbps m 20 M 1500
Port number: sender 2 receiver 39082 protocol 0
PATH rcvfrom: 10.1.13.2 (so-0/0/2.0) 17 pkts
Adspec: received MTU 1500
PATH sentto: localclient
RESV rcvfrom: localclient
  Record route: 10.1.36.2 10.1.13.2 <self>
Total 1 displayed, Up 1 , Down 0

```

Transit LSP: 0 sessions

Total 0 displayed, Up 0, Down 0

## Sample Output 2

### command-name

```

user@R3> show mpls lsp extensive
Ingress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 2 sessions

```

```

10.0.0.1
  From: 10.0.0.6, LSPstate: Up, ActiveRoute: 1
    LSPname: R6-to-R1 , LSPpath: Primary
  Suggested label received: -, Suggested label sent: -
  Recovery label received: -, Recovery label sent: 3
  Resv style: 1 FF, Label in: 100528, Label out: 3
  Time left: 125, Since: Thu Oct 21 15:29:26 2004
  Tspec: rate 0bps size 0bps peak Infbps m 20 M 1500
  Port number: sender 2 receiver 39082 protocol 0
  PATH rcvfrom: 10.1.36.2 (so-0/0/3.0) 17 pkts
  Adspec: received MTU 1500 sent MTU 1500
  PATH sentto: 10.1.13.1 (so-0/0/2.0) 17 pkts
  RESV rcvfrom: 10.1.13.1 (so-0/0/2.0) 17 pkts
  Explct route: 10.1.13.1
  Record route: 10.1.36.2 <self> 10.1.13.1

```

```

10.0.0.6
  From: 10.0.0.1, LSPstate: Up, ActiveRoute: 1
    LSPname: R1-to-R6 , LSPpath: Primary
  Suggested label received: -, Suggested label sent: -
  Recovery label received: -, Recovery label sent: 3
  Resv style: 1 FF, Label in: 100544, Label out: 3
  Time left: 147, Since: Thu Oct 21 15:39:33 2004
  Tspec: rate 0bps size 0bps peak Infbps m 20 M 1500
  Port number: sender 1 receiver 47963 protocol 0
  PATH rcvfrom: 10.1.13.1 (so-0/0/2.0) 4 pkts
  Adspec: received MTU 1500 sent MTU 1500
  PATH sentto: 10.1.36.2 (so-0/0/3.0) 4 pkts
  RESV rcvfrom: 10.1.36.2 (so-0/0/3.0) 4 pkts
  Explct route: 10.1.36.2
  Record route: 10.1.13.1 <self> 10.1.36.2
Total 2 displayed, Up 2, Down 0

```

### Sample Output 3

#### command-name

```

user@R6> show mpls lsp extensive
Ingress LSP: 1 sessions

```



10.0.0.1

From: 10.0.0.6, State: Up, ActiveRoute: 1, **LSPname: R6-to-R1**

ActivePath: (primary)

LoadBalance: Random

Encoding type: Packet, Switching type: Packet, GPID: IPv4

\*Primary State: Up

Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 20)

**10.1.36.1 S 10.1.13.1 S**

Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt):

10.1.36.1 10.1.13.1

18 Oct 21 15:34:18 Selected as active path

17 Oct 21 15:34:17 Record Route: 10.1.36.1 10.1.13.1

16 Oct 21 15:34:17 Up

15 Oct 21 15:34:17 Originate Call

14 Oct 21 15:34:17 CSPF: computation result accepted

[...Output truncated...]

Created: Tue Oct 19 22:28:30 2004

Total 1 displayed, Up 1, Down 0

Egress LSP: 1 sessions

10.0.0.6

From: 10.0.0.1, LSPstate: Up, ActiveRoute: 0

**LSPname: R1-to-R6**, LSPpath: Primary

Suggested label received: -, Suggested label sent: -

Recovery label received: -, Recovery label sent: -

Resv style: 1 FF, Label in: 3, Label out: -

Time left: 126, Since: Thu Oct 21 15:44:25 2004

Tspec: rate 0bps size 0bps peak Infbps m 20 M 1500

Port number: sender 1 receiver 47963 protocol 0

PATH rcvfrom: 10.1.36.1 (so-0/0/3.0) 4 pkts

Adspec: received MTU 1500

PATH sentto: localclient

RESV rcvfrom: localclient

**Record route: 10.1.13.1 10.1.36.1 <self>**

Total 1 displayed, Up 1, Down 0

Transit LSP: 0 sessions

Total 0 displayed, Up 0, Down 0

## Meaning

Sample Outputs 1 and 3 from ingress router R1 and egress router R6 show that the LSP is now traversing the network along the expected path, from R1 through R3 to R6, and the reverse LSP, from R6 through R3 to R1. In addition, Sample Output 2 from transit router R3 shows that there are two transit LSP sessions, one from R1 to R6, and the other from R6 to R1.

## Verifying the IS-IS Configuration on a Router in a Network

### IN THIS SECTION

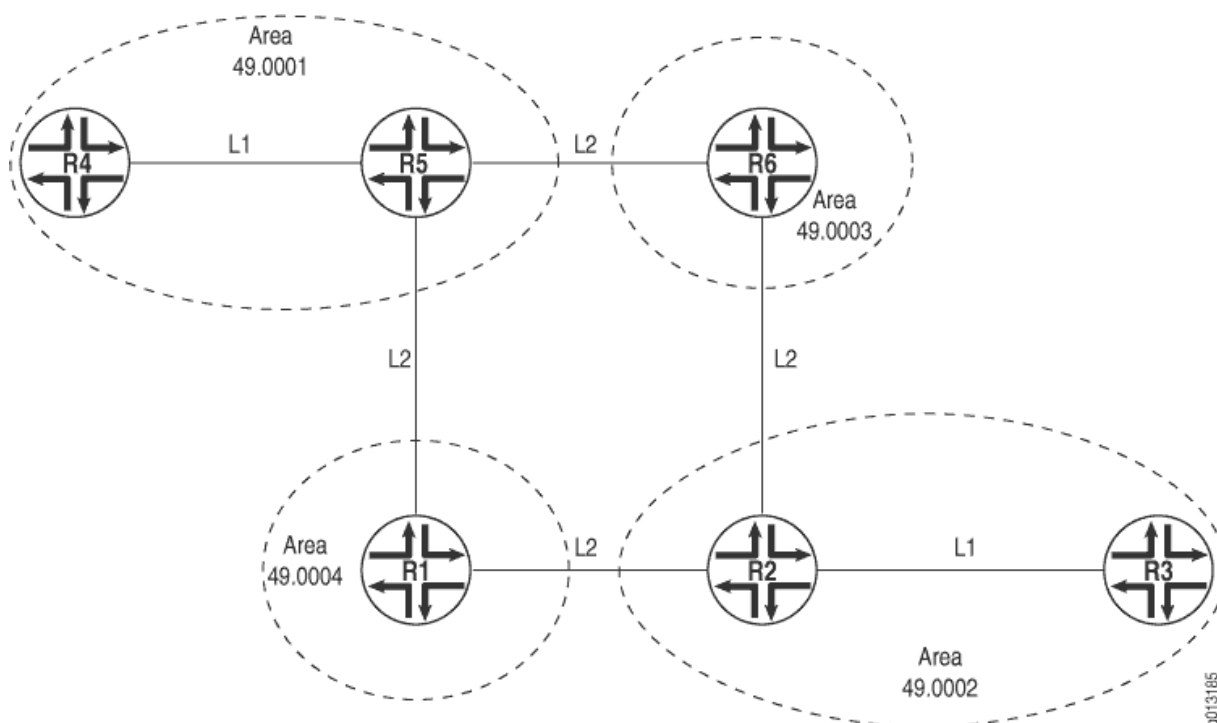
- [Check the Configuration of a Level 1/Level 2 Router | 782](#)
- [Check the Configuration of a Level 1 Router | 785](#)
- [Check the Configuration of a Level 2 Router | 788](#)

## Purpose

For IS-IS to run on a router (intermediate system) in your network, you must enable IS-IS on the router, configure a network entity title (NET) on the loopback interface (lo0), and configure `family iso` on all interfaces on which you want to run IS-IS. When you enable IS-IS on a router, Level 1 and Level 2 are enabled by default.

[Figure 55 on page 781](#) illustrates an example of routers at different levels in an IS-IS topology.

**Figure 55: Levels in an IS-IS Network Topology**

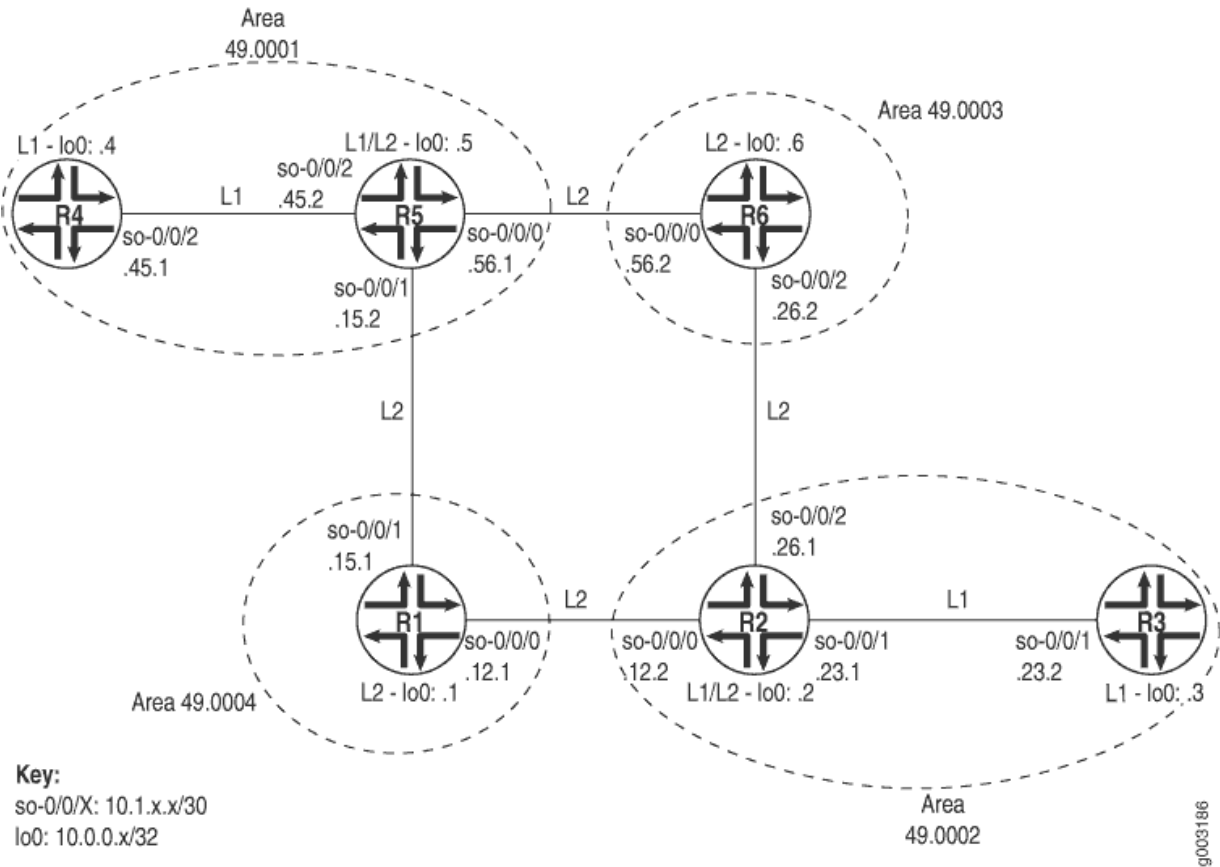


The network in [Figure 55 on page 781](#) is organized hierarchically and consists of Level 2, Level 1/Level 2, and Level 1 routers in one autonomous system (AS) divided into four areas: 49.0001, 49.0002, 49.0003, and 49.0004. The Level 2 routers route toward other autonomous systems. The Level 1/Level 2 routers route between areas and to other autonomous systems. The Level 1 routers route within an area, and when the destination is outside the local area, they route toward a Level1/Level2 system.

In the following topics, the configuration of the various types of routers is examined.

[Figure 56 on page 782](#) provides more details about the IS-IS network topology in [Figure 55 on page 781](#) so that you can verify the configuration output of the various routers.

Figure 56: IS-IS Network Topology with Details



To verify that IS-IS is configured correctly on routers at different levels, follow these steps:

### Check the Configuration of a Level 1/Level 2 Router

#### IN THIS SECTION

- Purpose | 782
- Action | 783
- Meaning | 785

#### Purpose

Check the configuration of a Level 1/Level 2 router.

## Action

To verify the IS-IS configuration of a Level 1/Level 2 router in your network, enter the following Junos OS command-line interface (CLI) commands:

```
user@host# [edit protocols isis] show
user@host# [edit protocols isis]
user@host# run show isis interface
user@host# [edit] edit interfaces
user@host# [edit interfaces] show
```

The following output is for an IS-IS configuration on R2, a Level 1/Level 2 router in the network shown.

## Sample Output

### command-name

```
[edit protocols isis]
user@R2# show
interface so-0/0/0.0 {
    level 2 metric 10;
    level 1 disable;
}
interface so-0/0/1.0 {
    level 2 disable;
    level 1 metric 10;
}
interface so-0/0/2.0 {
    level 2 metric 10;
    level 1 disable;
}
interface fxp0.0 {
    disable;
}
interface lo0.0;

[edit protocols isis]
user@R2# run show isis interface
IS-IS interface database:
Interface          L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0              0  0x1 Passive           Passive          0/0
```

```

so-0/0/0.0      2  0x1 Disabled      Point to Point      10/10
so-0/0/1.0      3  0x1 Point to Point Point to Point      10/10
so-0/0/2.0      2  0x1 Disabled      Point to Point      10/10
[edit interfaces]
user@R2# show
so-0/0/0 {
    unit 0 {
        family inet {
            address 10.1.12.2/30;
        }
        family iso;
    }
}
so-0/0/1 {
    unit 0 {
        family inet {
            address 10.1.23.1/30;
        }
        family iso;
    }
}
so-0/0/2 {
    unit 0 {
        family inet {
            address 10.1.26.1/30;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.0.0.2/32;
        }
        family iso {
            address 49.0002.1000.0000.0002.00;
        }
    }
}

```

## Meaning

The sample output shows a basic configuration of IS-IS on R2, a Level 1/Level 2 router. The basic configuration is at the [edit protocols isis] and [edit interfaces] hierarchy levels.

At the [edit protocols isis] level, five interfaces are included: so-0/0/0, so-0/0/1, so-0/0/2, fxp0, and the loopback interface (lo0). Two interfaces, so-0/0/0.0 and so-0/0/2.0, have Level 1 disabled, making them Level 2 interfaces. One interface, so-0/0/1.0, has Level 2 disabled, making it a Level 1 interface. The management interface (fxp0) is disabled so that IS-IS packets are not sent over it, and the loopback interface (lo0) is included because it becomes a point of connection from the router to the IS-IS network.

At the [edit interfaces] hierarchy level, all of the interfaces included in the [edit protocols isis] hierarchy level are configured with family iso, and the loopback interface (lo0) is configured with the NET address 49.0002.1000.0000.0002.00. Every router in an IS-IS network must have at least one NET address that identifies a point of connection to the IS-IS network. The NET address is generally configured on the loopback interface (lo0). Routers that participate in multiple areas can have multiple NET addresses.

## SEE ALSO

[Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding | 22](#)

## Check the Configuration of a Level 1 Router

### IN THIS SECTION

- [Purpose | 785](#)
- [Action | 786](#)
- [Meaning | 787](#)

## Purpose

To check the configuration of a Level 1 router.

## Action

To check the configuration of a Level 1 router, enter the following CLI commands:

```
user@host# [edit protocols isis] show
user@host# [edit protocols isis] run show isis interface
user@host# [edit] edit interfaces
user@host# [edit interfaces] show
```

The following sample output is for R4, a Level 1 router in the network shown in The following output is for an IS-IS configuration on R2, a Level 1/Level 2 router in the network shown.

## Sample Output

### command-name

```
[edit protocols isis]

user@R4# show
level 2 disable;
interface so-0/0/2.0 {
    level 1 metric 10;
}
interface fxp0.0 {
    disable;
}
interface lo0.0;
[edit protocols isis]

user@R4# run show isis interface
IS-IS interface database:
Interface          L  CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0              0   0x1 Passive           Passive           0/0
so-0/0/2.0         1   0x1 Point to Point    Disabled        10/10
[edit interfaces]

user@R4# show
so-0/0/2 {
    unit 0 {
        family inet {
            address 10.1.45.1/30;
        }
    }
}
```



```

        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.0.0.4/32;
        }
        family iso {
            address 49.0001.1000.0000.0004.00;
        }
    }
}
}

```

## Meaning

The sample output shows a basic configuration of IS-IS on R4, a Level 1 router. The basic configuration is at the [edit protocols isis] and [edit interfaces] hierarchy levels.

At the [edit protocols isis] hierarchy level, three interfaces are included: so-0/0/2.0, fxp0, and the loopback interface (lo0). Level 2 is disabled on the router, making it a Level 1 router that sends packets within its local area, 49.0001. When a packet destination is outside the local area, R4 establishes an adjacency with the nearest Level 1/Level 2 router (R5) that forwards the packets. For more information about adjacencies, see ["Displaying the Status of IS-IS Adjacencies " on page 790](#).

One interface, so-0/0/2.0, is configured for IS-IS. The management interface (fxp0) is disabled so that IS-IS packets are not sent over it, and the loopback interface (lo0) is included because it becomes a point of connection from the router to the IS-IS network.

At the [edit interfaces] hierarchy level, the interface included in the [edit protocols isis] hierarchy level is also configured with family iso, and the loopback interface (lo0) is configured with the NET address of 49.0001.1000.0000.0004.00. Every router in an IS-IS network must have at least one NET address that identifies a point of connection to the IS-IS network. The NET address is generally configured on the loopback interface (lo0). Routers that participate in multiple areas can have multiple NET addresses.

## SEE ALSO

[Example: Configuring IS-IS | 15](#)

## Check the Configuration of a Level 2 Router

### IN THIS SECTION

- Purpose | 788
- Action | 788
- Meaning | 790

### Purpose

Check the configuration of a Level 2 router.

### Action

To check the configuration of a Level 2 router, enter the following CLI commands:

```
user@host# [edit protocols isis] show
user@host# [edit protocols isis] run show isis interface
user@host# [edit] edit interfaces
user@host# [edit interfaces] show
```

The following sample output is for R6, a Level 2 router in the network shown.

### Sample Output

#### command-name

```
[edit protocols isis]
user@R6# show
level 1 disable;
interface so-0/0/0.0 {
    level 2 metric 10;
}
interface so-0/0/2.0 {
    level 2 metric 10;
}
interface fxp0.0 {
```

```

    disable;
}
interface lo0.0;

[edit protocols isis]
user@R6# run show isis interface
IS-IS interface database:
Interface          L CirID Level 1 DR      Level 2 DR      L1/L2 Metric
lo0.0              0  0x1 Passive           Passive          0/0
so-0/0/0.0         2  0x1 Disabled         Point to Point   10/10
so-0/0/2.0         2  0x1 Disabled         Point to Point   10/10

[edit interfaces]
user@R6# show
so-0/0/0 {
    unit 0 {
        family inet {
            address 10.1.56.2/30;
        }
        family iso;
    }
}
so-0/0/2 {
    unit 0 {
        family inet {
            address 10.1.26.2/30;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.0.0.6/32;
        }
        family iso {
            address 49.0003.1000.0000.0006.00;
        }
    }
}

```

## Meaning

The sample output shows a basic configuration of IS-IS on R6, a Level 2 router. The basic configuration is at the [edit protocols isis] and [edit interfaces] hierarchy levels.

At the [edit protocols isis] level, four interfaces are included: so-0/0/0.0, so-0/0/2.0, fxp0, and the loopback interface (lo0). Level 1 is disabled on the two SONET/SDH interfaces, making this a Level 2 router that routes between areas and toward other ASs. The management interface (fxp0) is disabled so that IS-IS packets are not sent over it, and the loopback interface (lo0) is included because it becomes a point of connection from the router to the IS-IS network.

At the [edit interfaces] hierarchy level, the interfaces included in the [edit protocols isis] hierarchy level are also configured with family iso, and the loopback interface (lo0) is configured with the NET address of 49.0003.1000.0000.0006.00. Every router in an IS-IS network must have at least one NET address that identifies a point of connection to the IS-IS network. The NET address is generally configured on the loopback interface (lo0). Routers that participate in multiple areas can have multiple NET addresses.

## SEE ALSO

[Example: Configuring IS-IS | 15](#)

## RELATED DOCUMENTATION

[Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups | 21](#)

[Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding | 22](#)

# Displaying the Status of IS-IS Adjacencies

## IN THIS SECTION

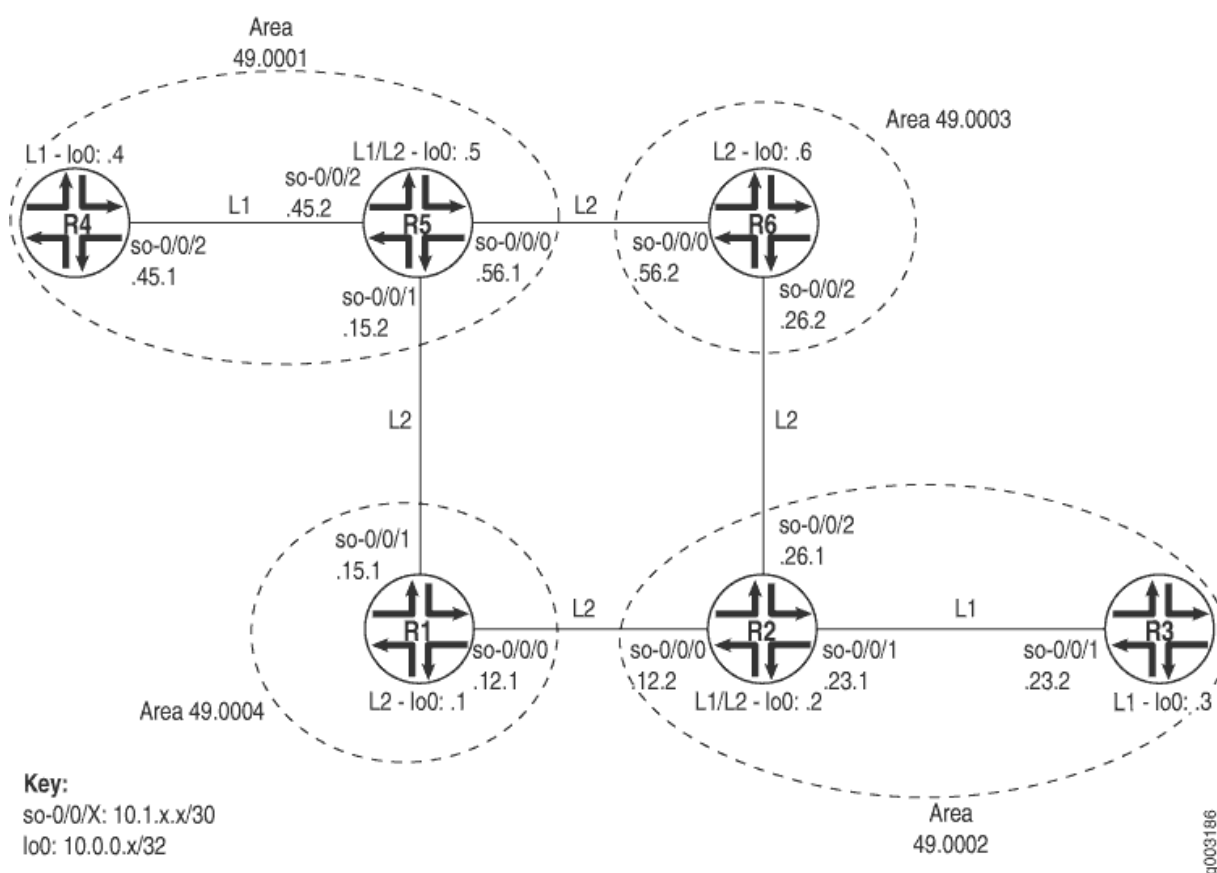
- [Verifying Adjacent Routers | 792](#)
- [Examine the Forwarding Table | 794](#)

## Purpose

Assuming that all the routers are correctly configured for IS-IS, you can verify which neighbors are adjacent and able to exchange IS-IS data. In addition, you can examine the set of routes installed in the forwarding table to verify that the routing protocol process (rpd) has relayed the correct information into the forwarding table.

Figure 57 on page 791 illustrates the example IS-IS topology used for the procedures in this topic.

**Figure 57: IS-IS Network Topology**



The network consists of Level 1 and Level 2 adjacencies. Level 1 adjacencies are within areas 49.0001 and 49.0002. Level 2 adjacencies occur between all directly connected Level 2 routers regardless of which area they are in. For example, R5 is in area 49.0001, R6 is in area 49.0003, R1 is in area 49.0004, and R2 is in area 49.0002. The network in Figure 57 on page 791 should have the following adjacencies:

- Level 2 adjacencies between all directly connected Level 2 routers (R1, R2, R5, and R6).
- Level 1 adjacencies between routers in area 49.0001 (R4 and R5) and between routers in area 49.0002 (R2 and R3).

To verify that routers are adjacent and able to exchange IS-IS data, follow these steps:

# Verifying Adjacent Routers

## IN THIS SECTION

- Purpose | 792
- Action | 792
- Meaning | 793

### Purpose

Verify that routers are adjacent and able to exchange IS-IS data.

### Action

To verify that routers are adjacent and able to exchange IS-IS data, enter the following CLI operational mode command:

```
user@host>
show isis adjacency
```

The following sample output shows the adjacencies that formed for all routers shown in [Displaying the Status of IS-IS Adjacencies](#) .

### Sample Output

#### command-name

```
user@R1> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/0.0     R2          2 Up         19
so-0/0/1.0     R5          2 Up         18

user@R2> show isis adjacency
Interface      System      L State      Hold (secs) SNPA
so-0/0/0.0     R1          2 Up         19
so-0/0/1.0     R3          1 Up         26
```

```

so-0/0/2.0          R6          2 Up          21

user@R3> show isis adjacency
Interface          System      L State      Hold (secs) SNPA
so-0/0/1.0          R2          1 Up          24

user@R4> show isis adjacency
Interface          System      L State      Hold (secs) SNPA
so-0/0/2.0          R5          1 Up          23

user@R5> show isis adjacency
Interface          System      L State      Hold (secs) SNPA
so-0/0/0.0          R6          2 Up          22
so-0/0/1.0          R1          2 Up          20
so-0/0/2.0          R4          1 Up          20

user@R6> show isis adjacency
Interface          System      L State      Hold (secs) SNPA
so-0/0/0.0          R5          2 Up          21
so-0/0/2.0          R2          2 Up          20

```

## Meaning

The sample output shows the adjacencies that formed in the network illustrated in [Displaying the Status of IS-IS Adjacencies](#). The Level 1/Level 2 routers (R2 and R5) formed Level 1 adjacencies with Level 1 routers (R3 and R4), and Level 2 adjacencies with the Level 2 routers (R1 and R6). To view the status of the adjacency, examine the State column. In this example, all adjacencies in the network are up.

If the state is not Up for a particular neighbor, you must first examine the IS-IS configuration for the particular interface. Make sure that the NET address is correct and that the loopback interface (lo0) is configured. Use the `show isis interface` or `show isis interface detail` command to display the IS-IS parameters for all interfaces configured with IS-IS. With these two commands, you can see which interfaces are configured for IS-IS, whether they are configured for Level 1 or Level 2, the IS-IS metric, and other IS-IS information.

## SEE ALSO

| [Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding](#)

## Examine the Forwarding Table

### IN THIS SECTION

- Purpose | 794
- Action | 794
- Meaning | 795

### Purpose

You can display the set of routes installed in the forwarding table to verify that the routing protocol process (rpd) has relayed the correct information into the forwarding table. This is especially important when there are network problems, such as connectivity. In this procedure, you verify that the routes displayed in Step 2 appear in the forwarding table for Router R5.

### Action

To examine the forwarding table for a router, enter the following CLI command:

```
user@host>
show route forwarding-table destination destination-prefix
```

### Sample Output

#### command-name

```
user@R5> show route forwarding-table destination
10.0.0.3
Routing table: inet
Internet:
Destination      Type RtRef Next hop          Type Index NhRef Netif
10.0.0.3/32      user   0 10.1.15.0          ucst  285   7 so-0/0/1.0
user@R5> show route forwarding-table destination 10.0.0.3
Routing table: inet
Internet:
Destination      Type RtRef Next hop          Type Index NhRef Netif
```



10.0.0.3/32	user	0	10.1.56.0	ucst	281	9	so-0/0/0.0
-------------	------	---	-----------	------	-----	---	------------

Meaning

The sample output shows the selected next hop between Routers R5 and R3 sent from the inet routing table and installed into the forwarding table. The first instance shows the route through Router R1, and the second instance shows the route through Router R6. In both instances, the preferred route displayed in Step 2 is installed in the forwarding table.

In general, the sample output includes the destination address and destination type, the next-hop address and next-hop type, the number of references to the next hop, an index number into an internal next-hop database, and the interface used to reach the next hop.

SEE ALSO

| [Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups](#)

RELATED DOCUMENTATION

| [Verifying the IS-IS Protocol](#) | 766

Displaying Detailed IS-IS Protocol Information

IN THIS SECTION

- [Action](#) | 795
- [Meaning](#) | 797

Action

To trace IS-IS messages in detail, follow these steps:

1. Configure the flag to display detailed IS-IS protocol messages.

```
[edit protocols isis traceoptions]
user@host# set flag hello detail
```

2. Verify the configuration.

```
user@host# show
```

For example:

```
[edit protocols isis traceoptions]
user@host# show
file isislog size 10k files 10;
flag hello detail;
```

3. Commit the configuration.

```
user@host# commit
```

4. View the contents of the file containing the detailed messages.

```
user@host# run show log filename
```

For example:

```
user@host# run show log isislog
```

```
Nov 29 23:17:50 trace_on: Tracing to "/var/log/isislog" started
Nov 29 23:17:50 Sending PTP IIH on so-1/1/1.0
Nov 29 23:17:53 Sending PTP IIH on so-1/1/0.0
Nov 29 23:17:54 Received PTP IIH, source id abc-core-01 on so-1/1/0.0
Nov 29 23:17:54   from interface index 11
Nov 29 23:17:54   max area 0, circuit type l2, packet length 4469
Nov 29 23:17:54   hold time 30, circuit id 6
Nov 29 23:17:54   neighbor state up
```

```

Nov 29 23:17:54 speaks IP
Nov 29 23:17:54 area address 99.0008 (1)
Nov 29 23:17:54 IP address 10.10.10.29
Nov 29 23:17:54 4396 bytes of total padding
Nov 29 23:17:54 updating neighbor abc-core-01
Nov 29 23:17:55 Received PTP IIH, source id abc-core-02 on so-1/1/1.0
Nov 29 23:17:55 from interface index 12
Nov 29 23:17:55 max area 0, circuit type 12, packet length 4469
Nov 29 23:17:55 hold time 30, circuit id 6
Nov 29 23:17:55 neighbor state up
Nov 29 23:17:55 speaks IP
Nov 29 23:17:55 area address 99.0000 (1)
Nov 29 23:17:55 IP address 10.10.10.33
Nov 29 23:17:55 4396 bytes of total padding
Nov 29 23:17:55 updating neighbor abc-core-02

```

## Meaning

Table 10 on page 797 lists tracing flags that can be configured specific to IS-IS and presents example output for some of the flags.

**Table 10: IS-IS Protocol Tracing Flags**

Tracing Flags	Description	Example Output
<b>csn</b>	Complete sequence number PDU (CSNP)	<p>Nov 28 20:02:48 Sending L2 CSN on interface so-1/1/0.0Nov 28 20:02:48 Sending L2 CSN on interface so-1/1/1.0</p> <p>With the <b>detail</b> option.</p> <p>Nov 28 20:06:08 Sending L2 CSN on interface so-1/1/1.0Nov 28 20:06:08 LSP abc-core-01.00-00 lifetime 1146Nov 28 20:06:08 sequence 0x1c4f8 checksum 0xa1e9Nov 28 20:06:08 LSP abc-core-02.00-00 lifetime 411Nov 28 20:06:08 sequence 0x7435 checksum 0x5424Nov 28 20:06:08 LSP abc-brdr-01.00-00 lifetime 465Nov 28 20:06:08 sequence 0xf73 checksum 0xab10Nov 28 20:06:08 LSP abc-edge-01.00-00 lifetime 1089Nov 28 20:06:08 sequence 0x1616 checksum 0xdb29Nov 28 20:06:08 LSP abc-edge-02.00-00 lifetime 1103Nov 28 20:06:08 sequence 0x45cc checksum 0x6883</p>

Table 10: IS-IS Protocol Tracing Flags *(Continued)*

Tracing Flags	Description	Example Output
<b>hello</b>	Hello packet	Nov 28 20:13:50 Sending PTP IIH on so-1/1/1.0Nov 28 20:13:50 Received PTP IIH, source id abc-core-01 on so-1/1/0.0Nov 28 20:13:53 Received PTP IIH, source id abc-core-02 on so-1/1/1.0Nov 28 20:13:57 Sending PTP IIH on so-1/1/0.0Nov 28 20:13:58 Received PTP IIH, source id abc-core-01 on so-1/1/0.0Nov 28 20:13:59 Sending PTP IIH on so-1/1/1.0
<b>lsp</b>	Link-state PDUs (LSPs)	Nov 28 20:15:46 Received L2 LSP abc-edge-01.00-00, interface so-1/1/0.0Nov 28 20:15:46 from abc-core-01Nov 28 20:15:46 sequence 0x1617, checksum 0xd92a, lifetime 1197Nov 28 20:15:46 Updating L2 LSP abc-edge-01.00-00 in TEDNov 28 20:15:47 Received L2 LSP abc-edge-01.00-00, interface so-1/1/1.0Nov 28 20:15:47 from abc-core-02Nov 28 20:15:47 sequence 0x1617, checksum 0xd92a, lifetime 1197
<b>lsp-generation</b>	Link-state PDU generation packets	Nov 28 20:21:24 Regenerating L1 LSP abc-edge-03.00-00, old sequence 0x682Nov 28 20:21:27 Rebuilding L1, fragment abc-edge-03.00-00Nov 28 20:21:27 Rebuilt L1 fragment abc-edge-03.00-00, size 59Nov 28 20:31:52 Regenerating L2 LSP abc-edge-03.00-00, old sequence 0x689Nov 28 20:31:54 Rebuilding L2, fragment abc-edge-03.00-00Nov 28 20:31:54 Rebuilt L2 fragment abc-edge-03.00-00, size 256Nov 28 20:34:05 Regenerating L1 LSP abc-edge-03.00-00, old sequence 0x683Nov 28 20:34:08 Rebuilding L1, fragment abc-edge-03.00-00Nov 28 20:34:08 Rebuilt L1 fragment abc-edge-03.00-00, size 59
<b>packets</b>	All IS-IS protocol packets	Not available.

Table 10: IS-IS Protocol Tracing Flags *(Continued)*

Tracing Flags	Description	Example Output
<b>psn</b>	Partial sequence number PDU (PSNP) packets	<p>Nov 28 20:40:39 Received L2 PSN, source abc-core-01, interface so-1/1/0.0</p> <p>Nov 28 20:40:39 Received L2 PSN, source abc-core-02, interface so-1/1/1.0</p> <p>Nov 28 20:41:36 Sending L2 PSN on interface so-1/1/1.0</p> <p>Nov 28 20:41:36 Sending L2 PSN on interface so-1/1/0.0</p> <p>Nov 28 20:42:35 Received L2 PSN, source abc-core-02, interface so-1/1/1.0</p> <p>Nov 28 20:42:35 LSP abc-edge-03.00-00 lifetime 1196</p> <p>Nov 28 20:42:35 sequence 0x68c checksum 0x746d</p> <p>Nov 28 20:42:35 Received L2 PSN, source abc-core-01, interface so-1/1/0.0</p> <p>Nov 28 20:42:35 LSP abc-edge-03.00-00 lifetime 1196</p> <p>Nov 28 20:42:35 sequence 0x68c checksum 0x746d</p> <p>Nov 28 20:42:49 Sending L2 PSN on interface so-1/1/1.0</p> <p>Nov 28 20:42:49 LSP abc-core-01.00-00 lifetime 1197</p> <p>Nov 28 20:42:49 sequence 0x1c4fb checksum 0x9bec</p> <p>Nov 28 20:42:49 Sending L2 PSN on interface so-1/1/0.0</p> <p>Nov 28 20:42:49 LSP abc-core-01.00-00 lifetime 1197</p> <p>Nov 28 20:42:49 sequence 0x1c4fb checksum 0x9bec</p>
<b>spf</b>	Shortest-path-first (SPF) calculations	<p>Nov 28 20:44:01 Scheduling SPF for L1: Reconfig</p> <p>Nov 28 20:44:01 Scheduling multicast SPF for L1: Reconfig</p> <p>Nov 28 20:44:01 Scheduling SPF for L2: Reconfig</p> <p>Nov 28 20:44:01 Scheduling multicast SPF for L2: Reconfig</p> <p>Nov 28 20:44:02 Running L1 SPF</p> <p>Nov 28 20:44:02 L1 SPF initialization complete: 0.000099s cumulative time</p> <p>Nov 28 20:44:02 L1 SPF primary processing complete: 0.000303s cumulative time</p> <p>Nov 28 20:44:02 L1 SPF result postprocessing complete: 0.000497s cumulative time</p> <p>Nov 28 20:44:02 L1 SPF RIB postprocessing complete: 0.000626s cumulative time</p> <p>Nov 28 20:44:02 L1 SPF routing table postprocessing complete: 0.000736s cumulative time</p>

## RELATED DOCUMENTATION

[Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups | 21](#)

[Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding | 22](#)

## Analyzing IS-IS Link-State PDUs in Detail

To analyze IS-IS link-state PDUs in detail, follow these steps:

1. Configure IS-IS open messages.

```
[edit protocols isis traceoptions]
user@host# set flag lsp detail
```

2. Verify the configuration.

```
user@host# show
```

For example:

```
[edit protocols isis traceoptions]
user@host# show
file isislog size 5m world-readable;
flag error;
flag lsp detail;
```

3. Commit the configuration.

```
user@host# commit
```

4. View the contents of the file containing the detailed messages.

```
user@host# run show log filename
```

For example:

```
user@host# run show log isislog
Nov 28 20:17:24 Received L2 LSP abc-core-01.00-00, interface so-1/1/0.0
Nov 28 20:17:24      from abc-core-01
Nov 28 20:17:24      sequence 0x1c4f9, checksum 0x9fea, lifetime 1199
Nov 28 20:17:24      max area 0, length 426
Nov 28 20:17:24      no partition repair, no database overload
Nov 28 20:17:24      IS type 3, metric type 0
Nov 28 20:17:24      area address 99.0908 (1)
Nov 28 20:17:24      speaks CLNP
Nov 28 20:17:24      speaks IP
Nov 28 20:17:24      dyn hostname abc-core-01
```

```

Nov 28 20:17:24      IP address 10.10.134.11
Nov 28 20:17:24      IP prefix: 10.10.10.0/30 metric 1 up
Nov 28 20:17:24      IP prefix: 10.10.10.4/30 metric 5 up
Nov 28 20:17:24      IP prefix: 10.10.10.56/30 metric 5 up
Nov 28 20:17:24      IP prefix: 10.10.10.52/30 metric 1 up
Nov 28 20:17:24      IP prefix: 10.10.10.64/30 metric 5 up
Nov 28 20:17:24      IP prefix: 10.10.10.20/30 metric 5 up
Nov 28 20:17:24      IP prefix: 10.10.10.28/30 metric 5 up
Nov 28 20:17:24      IP prefix: 10.10.10.44/30 metric 5 up
Nov 28 20:17:24      IP prefix 10.10.10.0 255.255.255.252
Nov 28 20:17:24      internal, metrics: default 1
Nov 28 20:17:24      IP prefix 10.10.10.4 255.255.255.252
Nov 28 20:17:24      internal, metrics: default 5
Nov 28 20:17:24      IP prefix 10.10.10.56 255.255.255.252
Nov 28 20:17:24      internal, metrics: default 5
Nov 28 20:17:24      IP prefix 10.10.10.52 255.255.255.252
Nov 28 20:17:24      internal, metrics: default 1
Nov 28 20:17:24      IP prefix 10.10.10.64 255.255.255.252
Nov 28 20:17:24      internal, metrics: default 5
Nov 28 20:17:24      IP prefix 10.10.10.20 255.255.255.252
Nov 28 20:17:24      internal, metrics: default 5
Nov 28 20:17:24      IP prefix 10.10.10.28 255.255.255.252
Nov 28 20:17:24      internal, metrics: default 5
Nov 28 20:17:24      IP prefix 10.10.10.44 255.255.255.252
Nov 28 20:17:24      internal, metrics: default 5
Nov 28 20:17:24      IS neighbors:
Nov 28 20:17:24      IS neighbor abc-core-02.00
Nov 28 20:17:24      internal, metrics: default 1
[...Output truncated...]
Nov 28 20:17:24      internal, metrics: default 5
Nov 28 20:17:24      IS neighbor abc-brdr-01.00
Nov 28 20:17:24      internal, metrics: default 5
Nov 28 20:17:24      IS neighbor abc-core-02.00, metric: 1
Nov 28 20:17:24      IS neighbor abc-esr-02.00, metric: 5
Nov 28 20:17:24      IS neighbor abc-edge-03.00, metric: 5
Nov 28 20:17:24      IS neighbor abc-edge-01.00, metric: 5
Nov 28 20:17:24      IS neighbor abc-edge-02.00, metric: 5
Nov 28 20:17:24      IS neighbor abc-brdr-01.00, metric: 5
Nov 28 20:17:24      IP prefix: 10.10.134.11/32 metric 0 up
Nov 28 20:17:24      IP prefix: 10.11.0.0/16 metric 5 up
Nov 28 20:17:24      IP prefix: 10.211.0.0/16 metric 0 up
Nov 28 20:17:24      IP prefix 10.10.134.11 255.255.255.255
Nov 28 20:17:24      internal, metrics: default 0

```

```

Nov 28 20:17:24      IP prefix 10.11.0.0 255.255.0.0
Nov 28 20:17:24      internal, metrics: default 5
Nov 28 20:17:24      IP prefix 10.211.0.0 255.255.0.0
Nov 28 20:17:24      internal, metrics: default 0
Nov 28 20:17:24      Updating LSP
Nov 28 20:17:24 Updating L2 LSP abc-core-01.00-00 in TED
Nov 28 20:17:24 Analyzing subtlv's for abc-core-02.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-esr-02.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-edge-03.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-edge-01.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-edge-02.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24 Analyzing subtlv's for abc-brdr-01.00
Nov 28 20:17:24 Analysis complete
Nov 28 20:17:24      Scheduling L2 LSP abc-core-01.00-00 sequence 0x1c4f9 on interface
so-1/1/1.0

```

## RELATED DOCUMENTATION

[Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups | 21](#)

[Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding | 22](#)

## Displaying Sent or Received IS-IS Protocol Packets

To configure the tracing for only sent or received IS-IS protocol packets, follow these steps:

1. Configure the flag to display sent, received, or both sent and received packets.

```

[edit protocols isis traceoptions]
user@host# set flag hello send

```



or

```
[edit protocols isis traceoptions]  
user@host# set flag hello receive
```

or

```
[edit protocols isis traceoptions]  
user@host# set flag hello
```

## 2. Verify the configuration.

```
user@host# show
```

For example:

```
[edit protocols isis traceoptions]  
user@host# show  
file isislog size 10k files 10;  
flag hello send;
```

or

```
[edit protocols isis traceoptions]  
user@host# show  
file isislog size 10k files 10;  
flag hello receive;
```

or

```
[edit protocols isis traceoptions]  
user@host# show  
file isislog size 10k files 10;  
flag hello send receive;
```

### 3. Commit the configuration.

```
user@host# commit
```

### 4. View the contents of the file containing the detailed messages.

```
user@host# run show log filename
```

For example:

```
user@host# run show log isislog
Sep 27 18:17:01 ISIS periodic xmit to 01:80:c2:00:00:15 (IFL 2)
Sep 27 18:17:01 ISIS periodic xmit to 01:80:c2:00:00:14 (IFL 2)
Sep 27 18:17:03 ISIS periodic xmit to 01:80:c2:00:00:15 (IFL 2)
Sep 27 18:17:04 ISIS periodic xmit to 01:80:c2:00:00:14 (IFL 2)
Sep 27 18:17:06 ISIS L2 hello from 0000.0000.0008 (IFL 2) absorbed
Sep 27 18:17:06 ISIS periodic xmit to 01:80:c2:00:00:15 (IFL 2)
Sep 27 18:17:06 ISIS L1 hello from 0000.0000.0008 (IFL 2) absorbed
```

## RELATED DOCUMENTATION

[Understanding IS-IS Areas to Divide an Autonomous System into Smaller Groups | 21](#)

[Example: Configuring a Multi-Level IS-IS Topology to Control Interarea Flooding | 22](#)

# 4

PART

## Configuration Statements and Operational Commands

---

- [Junos CLI Reference Overview | 806](#)
-

# Junos CLI Reference Overview

We've consolidated all Junos CLI commands and configuration statements in one place. Read this guide to learn about the syntax and options that make up the statements and commands. Also understand the contexts in which you'll use these CLI elements in your network configurations and operations.

- [Junos CLI Reference](#)

Click the links to access Junos OS and Junos OS Evolved configuration statement and command summary topics.

- [Configuration Statements](#)
- [Operational Commands](#)