



Junos[®] OS

Multitopology Routing Configuration Guide

Release

13.1



Published: 2013-02-14

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, 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.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

Junos® OS Multitopology Routing Configuration Guide

13.1

Copyright © 2013, 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 <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

| | | |
|------------------|--|-----------|
| | About the Documentation | ix |
| | Documentation and Release Notes | ix |
| | Supported Platforms | ix |
| | Using the Examples in This Manual | ix |
| | Merging a Full Example | x |
| | Merging a Snippet | x |
| | Documentation Conventions | xi |
| | Documentation Feedback | xiii |
| | Requesting Technical Support | xiii |
| | Self-Help Online Tools and Resources | xiii |
| | Opening a Case with JTAC | xiv |
| Part 1 | Overview | |
| Chapter 1 | Introduction to Multitopology Routing | 3 |
| | Multitopology Routing Overview | 4 |
| | Routing Table Naming Conventions for Multitopology Routing | 6 |
| | Filter-Based Forwarding Support | 7 |
| Chapter 2 | Multitopology Routing Standards | 9 |
| | Multitopology Routing Standards | 9 |
| Part 2 | Configuration | |
| Chapter 3 | Multitopology Routing | 13 |
| | Example: Configuring Multitopology Routing Based on Applications | 13 |
| | Understanding Multitopology Routing Based on Applications | 13 |
| | Example: Configuring Multitopology Routing Based on Applications | 15 |
| | Example: Configuring Multitopology Routing Based on a Multicast Source | 39 |
| | Understanding Multitopology Routing Based on a Multicast Source | 39 |
| | Example: Configuring Multitopology Routing Based on a Multicast Source | 41 |
| Chapter 4 | Multitopology Routing Configuration Statements | 67 |
| | community (Protocols BGP) | 68 |
| | rib (Multitopology Routing) | 69 |
| | topologies (Multitopology Routing) | 70 |
| | topology (Filter-Based Forwarding) | 71 |
| | topology (Multitopology Routing) | 72 |
| | topology (OSPF) | 73 |
| | topology (OSPF Interface) | 74 |
| | topology-id | 75 |

| | | |
|------------------|---|------------|
| Part 3 | Administration | |
| Chapter 5 | Multitopology Routing Operational Commands | 79 |
| | show ospf database | 80 |
| | show ospf3 database | 89 |
| | show (ospf ospf3) interface | 100 |
| | show (ospf ospf3) neighbor | 106 |
| | show route | 113 |
| | show route detail | 119 |
| | show route forwarding-table | 134 |
| | show route hidden | 148 |
| | show route summary | 151 |
| | show route table | 155 |
| | traceroute | 166 |
| Part 4 | Troubleshooting | |
| Chapter 6 | Routing Protocol Process Memory FAQs | 173 |
| | Routing Protocol Process Memory FAQs Overview | 173 |
| | Routing Protocol Process Memory FAQs | 174 |
| | Frequently Asked Questions: Routing Protocol Process Memory | 174 |
| | Frequently Asked Questions: Interpreting Routing Protocol Process-Related Command Outputs | 175 |
| | Frequently Asked Questions: Routing Protocol Process Memory Swapping | 178 |
| | Frequently Asked Questions: Troubleshooting the Routing Protocol Process | 179 |
| Part 5 | Index | |
| | Index | 183 |

List of Figures

| | | |
|------------------|---|-----------|
| Part 1 | Overview | |
| Chapter 1 | Introduction to Multitopology Routing | 3 |
| | Figure 1: Voice and Video Routing Topologies Enabled on a Subset of Links | 5 |
| Part 2 | Configuration | |
| Chapter 3 | Multitopology Routing | 13 |
| | Figure 2: MTR-OSPF Area Boundary | 14 |
| | Figure 3: BGP Route Resolution | 14 |
| | Figure 4: Route Resolution for MTR | 15 |
| | Figure 5: Multitopology OSPF and IBGP for Designating Links Belonging to Voice and Video Services | 16 |
| | Figure 6: Core Links Configured to Prefer Specified Routing Topologies | 39 |
| | Figure 7: Core Links Configured to Prefer Specified Routing Topologies | 41 |
| | Figure 8: Multi-topology OSPF and BGP for Designating Links Belonging to Voice and Video Services | 42 |

List of Tables

| | | |
|------------------|---|------------|
| | About the Documentation | ix |
| | Table 1: Notice Icons | xi |
| | Table 2: Text and Syntax Conventions | xi |
| Part 1 | Overview | |
| Chapter 1 | Introduction to Multitopology Routing | 3 |
| | Table 3: Examples of Routing Tables for Custom Topologies | 7 |
| Part 3 | Administration | |
| Chapter 5 | Multitopology Routing Operational Commands | 79 |
| | Table 4: show ospf database Output Fields | 81 |
| | Table 5: show ospf3 database Output Fields | 90 |
| | Table 6: show (ospf ospf3) interface Output Fields | 101 |
| | Table 7: show (ospf ospf3) neighbor Output Fields | 107 |
| | Table 8: show route Output Fields | 114 |
| | Table 9: show route detail Output Fields | 119 |
| | Table 10: Next-hop Types Output Field Values | 123 |
| | Table 11: State Output Field Values | 125 |
| | Table 12: Communities Output Field Values | 127 |
| | Table 13: show route forwarding-table Output Fields | 136 |
| | Table 14: show route summary Output Fields | 151 |
| | Table 15: traceroute Output Fields | 168 |
| Part 4 | Troubleshooting | |
| Chapter 6 | Routing Protocol Process Memory FAQs | 173 |
| | Table 16: show system processes extensive Output Fields | 176 |
| | Table 17: show task memory Output Fields | 177 |

About the Documentation

- Documentation and Release Notes on page ix
- Supported Platforms on page ix
- Using the Examples in This Manual on page ix
- Documentation Conventions on page xi
- Documentation Feedback on page xiii
- Requesting Technical Support on page xiii

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at <http://www.juniper.net/techpubs/>.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <http://www.juniper.net/books>.

Supported Platforms

For the features described in this document, the following platforms are supported:

- T Series
- MX Series
- M Series

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

```
system {
  scripts {
    commit {
      file ex-script.xml;
    }
  }
}
interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}
```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

```
[edit]
user@host# load merge /var/tmp/ex-script.conf
load complete
```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

```
commit {
  file ex-script-snippet.xml; }
```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see the CLI User Guide.

Documentation Conventions

Table 1 on page xi defines notice icons used in this guide.

Table 1: Notice Icons

| Icon | Meaning | Description |
|---|--------------------|---|
|  | Informational note | Indicates important features or instructions. |
|  | Caution | Indicates a situation that might result in loss of data or hardware damage. |
|  | Warning | Alerts you to the risk of personal injury or death. |
|  | Laser warning | Alerts you to the risk of personal injury from a laser. |

Table 2 on page xi defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

| Convention | Description | Examples |
|----------------------------|--|--|
| Bold text like this | Represents text that you type. | To enter configuration mode, type the configure command: user@host> configure |
| Fixed-width text like this | Represents output that appears on the terminal screen. | user@host> show chassis alarms No alarms currently active |

Table 2: Text and Syntax Conventions (*continued*)

| Convention | Description | Examples |
|--------------------------------|--|---|
| <i>Italic text like this</i> | <ul style="list-style-type: none"> Introduces or emphasizes important new terms. Identifies book names. Identifies RFC and Internet draft titles. | <ul style="list-style-type: none"> A policy <i>term</i> is a named structure that defines match conditions and actions. <i>Junos OS System Basics Configuration Guide</i> RFC 1997, <i>BGP Communities Attribute</i> |
| <i>Italic text like this</i> | Represents variables (options for which you substitute a value) in commands or configuration statements. | Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i> |
| Text like this | Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components. | <ul style="list-style-type: none"> To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level. The console port is labeled CONSOLE. |
| < > (angle brackets) | Enclose optional keywords or variables. | stub <default-metric metric>; |
| (pipe symbol) | Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity. | broadcast multicast <i>(string1 string2 string3)</i> |
| # (pound sign) | Indicates a comment specified on the same line as the configuration statement to which it applies. | rsvp { # Required for dynamic MPLS only |
| [] (square brackets) | Enclose a variable for which you can substitute one or more values. | community name members [community-ids] |
| Indentation and braces ({ }) | Identify a level in the configuration hierarchy. | [edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } } |
| ;(semicolon) | Identifies a leaf statement at a configuration hierarchy level. | |
| J-Web GUI Conventions | | |
| Bold text like this | Represents J-Web graphical user interface (GUI) items you click or select. | <ul style="list-style-type: none"> In the Logical Interfaces box, select All Interfaces. To cancel the configuration, click Cancel. |
| > (bold right angle bracket) | Separates levels in a hierarchy of J-Web selections. | In the configuration editor hierarchy, select Protocols>Ospf . |

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at <https://www.juniper.net/cgi-bin/docbugreport/>. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release version (if applicable)

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <http://www.juniper.net/support/warranty/>.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>
- Download the latest versions of software and review release notes: <http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <https://www.juniper.net/alerts/>

- Join and participate in the Juniper Networks Community Forum:
<http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://tools.juniper.net/SerialNumberEntitlementSearch/>

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/>.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <http://www.juniper.net/support/requesting-support.html>.

PART 1

Overview

- [Introduction to Multitopology Routing on page 3](#)
- [Multitopology Routing Standards on page 9](#)

CHAPTER 1

Introduction to Multitopology Routing

- [Multitopology Routing Overview on page 4](#)

Multitopology Routing Overview

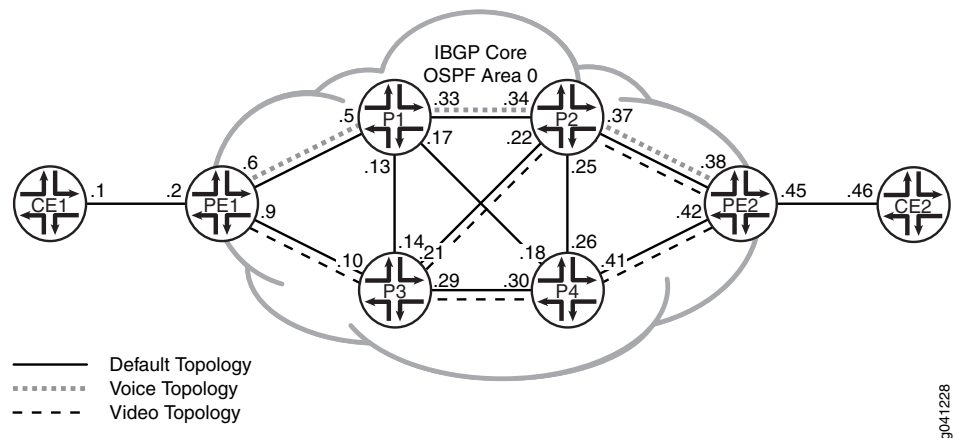
Service providers and enterprises can use multitopology routing (MTR) to engineer traffic flow across a network. MTR can be used with direct and static routes, OSPF, and BGP. MTR can be configured for unicast and multicast IP, using the Junos[®] operating system (Junos OS). With basic unicast IP, an IBGP core runs on top of OSPF to direct traffic based on application types, such as voice or video. For multicast, Protocol Independent Multicast (PIM) is used, in conjunction with multitopology OSPF and BGP, to direct multicast traffic over particular paths based on traffic characteristics.

In a network carrying multiple traffic types, you often need to direct different types of application traffic over multiple links depending on their link characteristics. For example, voice traffic requires links that are less likely to incur latency, jitter, or packet loss. File traffic, on the other hand, requires links that have large amounts of available bandwidth. MTR is a way to direct traffic to follow specified paths. You can use MTR to extend a traditional MPLS network into a segment where only IP forwarding is supported. With MTR, each traffic type is handled in its own conceptually incongruent topology, and yet runs on top of the same underlying network. You can configure separate topologies to share the same network links as needed. MTR uses a combination of control plane (routing) and forwarding plane (firewall filters). Each topology uses the unified control plane to make routing decisions for traffic associated with that topology. In addition, each topology has a separate forwarding table and, in effect, a dedicated forwarding plane for each topology. This forwarding plane not only directs traffic using its own forwarding table, but also simultaneously handles sophisticated functionality, such as queuing for class of service (CoS), that can be applied to a topology. As traffic enters the router, fields within a packet determine to which topology the traffic belongs.

Multitopology routing enables you to configure class-based forwarding for different types of traffic, such as voice, video, and data. Each type of traffic is defined by a topology that is used to create a new routing table for that topology. MTR provides the ability to generate forwarding tables based on the resolved entries in the routing tables for the custom topologies you create. In this way, packets of different classes can be routed independently from one another.

One way to manage traffic flow is to group links into specific routing topologies based on application requirements. Each routing topology can be thought of as a set of contiguous links. MTR provides a way for you to manage each set of links uniquely by directing traffic types to flow over specified links. This solution uses a combination of routing (control plane) and firewall filtering (forwarding plane) configurations. [Figure 1 on page 5](#) shows a network with two topologies configured: voice (dotted lines) and video (dashed lines). Note there is also a default routing topology (solid line).

Figure 1: Voice and Video Routing Topologies Enabled on a Subset of Links



g041228

You can configure MTR for BGP, OSPF, and static routes. When a routing topology is created, it has its own forwarding table.

Packet forwarding uses firewall filters to examine packets as they enter the router over an interface. These filters determine whether a specific topology or the default forwarding table should be used to make packet forwarding decisions. If applicable, firewall filters evaluate packet attributes, such as destination IP address, Differentiated Services code points (DSCPs), or next-level protocol headers, to determine which topology to use. In fact, any item in a packet that is recognized by a firewall filter can be used to direct the packet next-hop lookup to use a particular topology. Once the packet is directed to use a topology, the destination IP address must be in the topology forwarding table. Otherwise, the packet is dropped.

The following topics provide background information about multitopology routing:

- [Routing Table Naming Conventions for Multitopology Routing on page 6](#)
- [Filter-Based Forwarding Support on page 7](#)

Routing Table Naming Conventions for Multitopology Routing

Routing topologies have their own routing tables, similar to any other routing table created by default or by a **rib-group** configuration with a few differences. The routing table name indicates that the routing table is associated with a topology by prepending a colon (:) to the name. For example, a routing topology named **voice** has a routing table named **:voice.inet.0**. When routing topologies are configured under **routing-options**, a new routing table for each topology is created.

Each routing protocol creates a routing table based on the topology name, the instance name, and the purpose of the table. A routing table for each topology uses the following format:

logical-system-name/routing-instance-name:topology-name.protocol.identifier

The routing instance string is included only if the instance is not the master. The logical system string is included only if the logical system identifier has a value other than 0 (zero). Each routing table for a topology includes a colon (:) before the topology name that also separates the routing-instance name from the topology name. **protocol** is the

protocol family, which can be **inet** or **inet6**. *identifier* is a positive integer that specifies the instance of the routing table. [Table 3 on page 7](#) shows specific examples of routing tables for various topologies.

Table 3: Examples of Routing Tables for Custom Topologies

| Name of Routing Table | Description |
|--------------------------------------|--|
| <code>:voice.inet.0</code> | Master instance, voice topology, unicast IPv4 routes |
| <code>:voice.inet6.0</code> | Master instance, voice topology, unicast IPv6 routes |
| <code>:voice.inet.3</code> | Master instance, voice topology, ingress label-switched paths (LSPs) |
| <code>private_1/:voice.inet.0</code> | Logical system private, voice topology, unicast IPv4 routes |
| <code>customer-A:voice.inet.0</code> | Virtual-router customer-A, voice topology, unicast IPv4 routes |
| <code>customer-B:voice.inet.3</code> | Virtual-router customer-B, voice topology, ingress LSPs |
| <code>customer-A:voice.mpls.0</code> | Virtual-router customer-A, voice topology, unicast carrier-of-carriers IPv4 routes |

To run multitopology routing (MTR), you must configure IP routing. MTR supports OSPF version 2 (OSPFv2), static routes, and BGP. You must configure an interior gateway protocol (IGP), such as OSPFv2 or static routing. Configure BGP to add routes learned through BGP to the appropriate custom topologies.

MTR is also supported on logical systems and the virtual router routing instance. No other routing instance type is supported on MTR.

Filter-Based Forwarding Support

By default, the ingress interface forwards traffic to the default topology for each configured routing instance. MTR supports filter-based forwarding, which enables you to match traffic on the ingress interface with a specific type of forwarding class and then forward that traffic to the specified topology. You can further define how traffic is handled for each forwarding class by configuring additional firewall filters that match traffic for such values as the IP precedence field or the Differentiated Services code point (DSCP).

- Related Documentation**
- [Example: Configuring Multitopology Routing Based on Applications on page 13](#)
 - [Example: Configuring Multitopology Routing Based on a Multicast Source on page 39](#)

CHAPTER 2

Multitopology Routing Standards

- [Multitopology Routing Standards on page 9](#)

Multitopology Routing Standards

Multitopology routing is defined in the following document:

- RFC 4915, *Multi-Topology (MT) Routing in OSPF*

PART 2

Configuration

- [Multitopology Routing on page 13](#)
- [Multitopology Routing Configuration Statements on page 67](#)

CHAPTER 3

Multitopology Routing

- [Example: Configuring Multitopology Routing Based on Applications on page 13](#)
- [Example: Configuring Multitopology Routing Based on a Multicast Source on page 39](#)

Example: Configuring Multitopology Routing Based on Applications

- [Understanding Multitopology Routing Based on Applications on page 13](#)
- [Example: Configuring Multitopology Routing Based on Applications on page 15](#)

Understanding Multitopology Routing Based on Applications

Multitopology routing (MTR) enables you to configure class-based forwarding for different types of traffic, such as voice, video, and data. Each type of traffic is defined by a topology that is used to create a new routing table for that topology. MTR provides the ability to generate forwarding tables based on the resolved entries in the routing tables for the custom topologies you create. In this way, packets of different classes can be routed independently from one another.

To run MTR, you must configure IP routing. MTR supports OSPFv2, static routes, and BGP. You must configure an interior gateway protocol (IGP), such as OSPFv2 or static routing. Configure BGP to add routes learned through BGP to the appropriate custom topologies. MTR also supports filter-based forwarding, which enables you to match traffic on the ingress interface with a specific type of forwarding class and then forward that traffic to the specified topology.

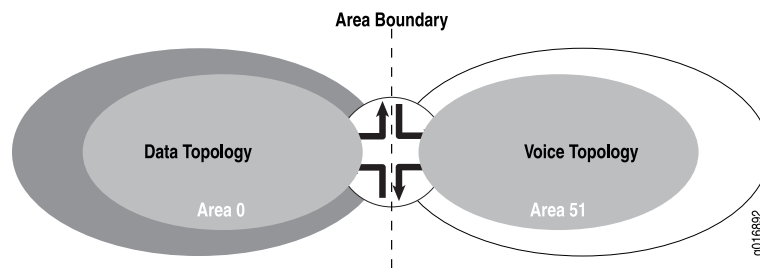
OSPF in MTR

OSPF in MTR uses a single instance of OSPF to carry connectivity and IP reachability information for different topologies. That information is used to calculate shortest-path-first (SPF) trees and routing tables. OSPF for MTR supports protocol extensions that include metrics that correspond to different topologies for link and prefix reachability information. The type-of-service (TOS) metric field is used to advertise the topology-specific metric for links and prefixes belonging to that topology. The TOS field is redefined as MT-ID in the payload of router, summary, and Type 5 and Type 7 AS-external link-state advertisements (LSAs).

Under MTR, each OSPF interface continues to belong to a single area. Therefore, by default, all topologies share the same area boundaries. As a result, attributes of an area, such as stubbiness, are independent of the topology. By default, all topologies configured

for OSPF are enabled on all interfaces. However, you can disable one or more configured topologies on an interface. You can thus allocate an interface for a specific topology. In [Figure 2 on page 14](#), Area 51 includes an interface that is uniquely allocated to voice traffic, and Area 0 includes an interface that is uniquely allocated to data traffic. Each topology thus corresponds to a different OSPF area that shares a boundary.

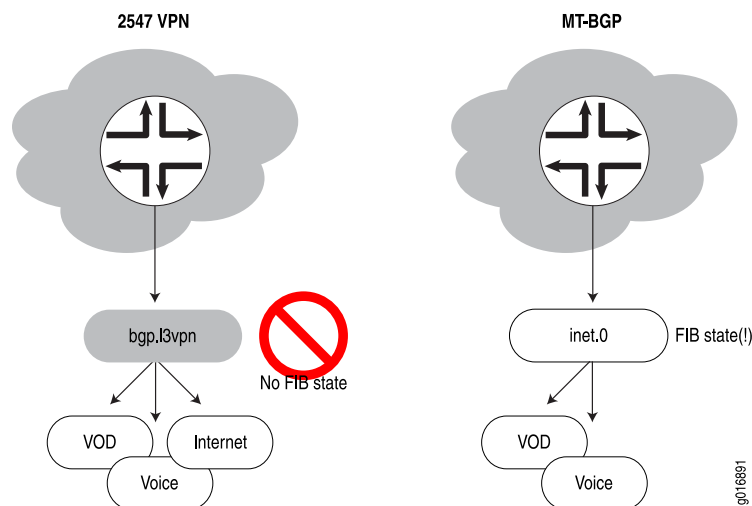
Figure 2: MTR-OSPF Area Boundary



BGP in MTR

BGP in MTR provides the ability to resolve BGP routes against configured topologies. An inbound policy is used to select routes for inclusion in the appropriate routing tables for the topologies. The default behavior for virtual private networks (VPNs) that use MPLS for forwarding packets over the backbone and that use BGP for distributing routes over the backbone is to place BGP route updates in the **bgp.l3vpn** routing table. [Figure 3 on page 14](#) shows a BGP peer operating in an environment that conforms with the requirements in RFC 2547, *BGP/MPLS VPNs*. The figure shows how a BGP peer configured for MTR performs secondary route resolution.

Figure 3: BGP Route Resolution

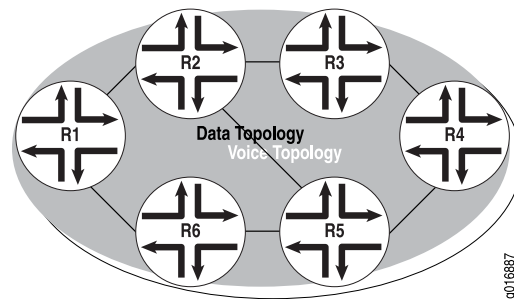


The BGP peer in a standard VPN topology places prefixes for routes it learns in the **bgp.l3vpn** routing table, which does not result in automatic updates to the forwarding table. Under BGP in MTR, when BGP receives a route from a peer, it attempts to resolve that route against a route in the **inet.0** routing table. If the route is resolved, it is placed in that table, which generates a forwarding state. If you have configured a community

target identifier that matches the import policy for the topology, routing and forwarding states are added to the tables for the topology.

Because MTR provides support for BGP to perform secondary route resolution, as [Figure 4 on page 15](#) shows, MTR is able to create two distinct network paths for each type of traffic. Each router advertises BGP routes that need to be resolved against the IGP routes for each topology. Based on the IGP metrics configured for each topology, for all routes that originate from Router 4 (R4), the upper path between R1 and R4, which traverses R2 and R3, is selected for voice traffic, whereas the lower path between R1 and R4, which traverses R5 and R6, is selected for data traffic.

Figure 4: Route Resolution for MTR



Example: Configuring Multitopology Routing Based on Applications

This example shows how to use multitopology routing (MTR) to choose a topology path based on an application, either voice or video.

- [Requirements on page 15](#)
- [Overview on page 15](#)
- [Configuration on page 16](#)
- [Verification on page 33](#)

Requirements

This example requires that Junos OS Release 9.0 or later is running on the provider core devices.

Overview

In this example, the network is running OSPF and internal BGP (IBGP) in the core, but not MPLS. Even without traffic engineering, voice traffic uses one set of links, and video traffic uses a different set of links. This traffic might or might not be destined for the same IP address. In some cases, both applications traverse the same link. The solution uses MTR-based OSPF and BGP, along with firewall filters to direct different traffic types over designated links. The routers use a fairly similar set of configurations, which reduces complexity and improves network management.

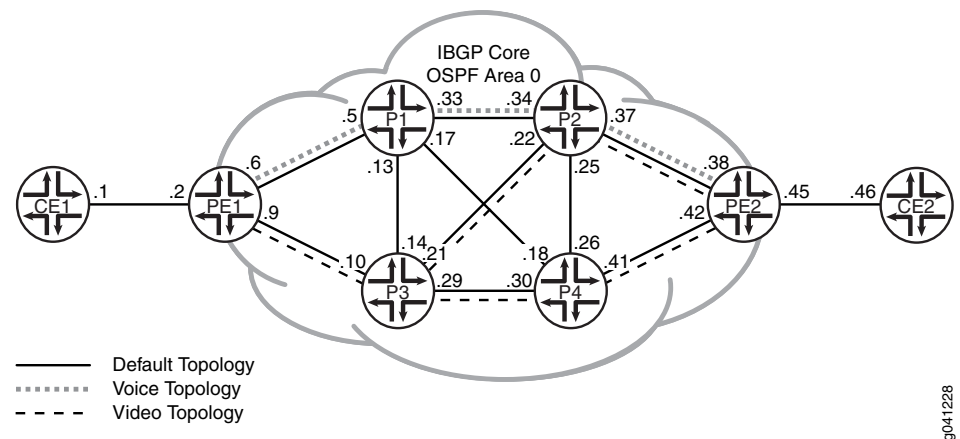
The OSPF topologies are defined to support each service offering over the OSPF area. The links of a topology must be contiguous, consistent with a typical OSPF area. IBGP routes in each routing topology automatically use the associated OSPF topology routing table for protocol next-hop route resolution. No special route resolution configurations

are required. In this solution, multiple topologies can be configured over the same link. However, traffic in each application service class cannot traverse links unless they are configured for the topology designated for that service. [Figure 5 on page 16](#) shows a diagram of this case. Contiguous paths for routing the voice topology are shown with dotted lines, and paths for routing the video topology are shown with dashed lines.

For a complete set of configurations for all of the devices in the topology, see “[CLI Quick Configuration](#)” on page 16. The remainder of the example focuses on Device CE1 and Device PE1.

[Figure 5 on page 16](#) shows the sample topology.

Figure 5: Multitopology OSPF and IBGP for Designating Links Belonging to Voice and Video Services



Configuration

- [Configuring Device CE1 on page 23](#)
- [Configuring Device PE1 on page 26](#)

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-0/1/0 fastether-options loopback
set interfaces fe-0/1/0 unit 0 family inet address 11.19.130.1/24
set interfaces fe-0/1/0 unit 0 family inet address 11.19.131.1/24
set interfaces fe-0/1/0 unit 0 family inet address 11.19.132.1/24
set interfaces fe-1/2/0 unit 1 description to-PE1
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces lo0 unit 97 family inet address 10.255.165.97/32 primary
set protocols bgp group ebgp type external
set protocols bgp group ebgp local-address 10.0.0.1
set protocols bgp group ebgp export set_community
set protocols bgp group ebgp export inject_directs
set protocols bgp group ebgp peer-as 100
set protocols bgp group ebgp neighbor 10.0.0.2
set policy-options policy-statement inject_directs term a from protocol direct
```

```

set policy-options policy-statement inject_directs term a from interface fe-0/1/0.0
set policy-options policy-statement inject_directs term a then next policy
set policy-options policy-statement inject_directs term a then accept
set policy-options policy-statement inject_directs term b then reject
set policy-options policy-statement set_community term a from route-filter 11.19.130.0/24
  exact
set policy-options policy-statement set_community term a from route-filter 11.19.131.0/24
  exact
set policy-options policy-statement set_community term a then community add voice
set policy-options policy-statement set_community term a then accept
set policy-options policy-statement set_community term b from route-filter 11.19.132.0/24
  exact
set policy-options policy-statement set_community term b from route-filter 11.19.133.0/24
  exact
set policy-options policy-statement set_community term b then community add video
set policy-options policy-statement set_community term b then accept
set policy-options policy-statement set_community term default then accept
set policy-options community video members target:50:50
set policy-options community voice members target:40:40
set routing-options autonomous-system 101

```

Device CE2

```

set interfaces fe-0/1/1 fastether-options loopback
set interfaces fe-0/1/1 unit 0 family inet address 11.19.140.1/24
set interfaces fe-0/1/1 unit 0 family inet address 11.19.141.1/24
set interfaces fe-0/1/1 unit 0 family inet address 11.19.142.1/24
set interfaces fe-1/2/0 unit 46 description to-PE2
set interfaces fe-1/2/0 unit 46 family inet address 10.0.0.46/30
set interfaces lo0 unit 20 family inet address 10.255.165.20/32 primary
set protocols bgp group ebgp type external
set protocols bgp group ebgp local-address 10.0.0.46
set protocols bgp group ebgp export set_community
set protocols bgp group ebgp export inject_directs
set protocols bgp group ebgp peer-as 100
set protocols bgp group ebgp neighbor 10.0.0.45
set policy-options policy-statement inject_directs term a from protocol direct
set policy-options policy-statement inject_directs term a from interface fe-0/1/1.0
set policy-options policy-statement inject_directs term a then next policy
set policy-options policy-statement inject_directs term a then accept
set policy-options policy-statement inject_directs term b then reject
set policy-options policy-statement set_community term a from route-filter 11.19.140.0/24
  exact
set policy-options policy-statement set_community term a from route-filter 11.19.141.0/24
  exact
set policy-options policy-statement set_community term a then community add voice
set policy-options policy-statement set_community term a then accept
set policy-options policy-statement set_community term b from route-filter 11.19.142.0/24
  exact
set policy-options policy-statement set_community term b from route-filter 11.19.143.0/24
  exact
set policy-options policy-statement set_community term b then community add video
set policy-options policy-statement set_community term b then accept
set policy-options policy-statement set_community term default then accept
set policy-options community video members target:50:50
set policy-options community voice members target:40:40
set routing-options autonomous-system 102

```

Device PE1

```
set interfaces fe-1/2/0 unit 2 description to-CE1
set interfaces fe-1/2/0 unit 2 family inet filter input ef_path
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 6 description to-P1
set interfaces fe-1/2/1 unit 6 family inet filter input ef_path
set interfaces fe-1/2/1 unit 6 family inet address 10.0.0.6/30
set interfaces fe-1/2/2 unit 9 description to-P3
set interfaces fe-1/2/2 unit 9 family inet filter input ef_path
set interfaces fe-1/2/2 unit 9 family inet address 10.0.0.9/30
set interfaces lo0 unit 93 family inet address 10.255.165.93/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.93
set protocols bgp group ibgp family inet unicast topology voice community target:40:40
set protocols bgp group ibgp family inet unicast topology video community target:50:50
set protocols bgp group ibgp export nhs
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.113
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols bgp group ebgp type external
set protocols bgp group ebgp local-address 10.0.0.2
set protocols bgp group ebgp family inet unicast topology voice community target:40:40
set protocols bgp group ebgp family inet unicast topology video community target:50:50
set protocols bgp group ebgp peer-as 101
set protocols bgp group ebgp neighbor 10.0.0.1
set protocols ospf topology voice topology-id 126
set protocols ospf topology video topology-id 52
set protocols ospf area 0.0.0.0 interface fe-1/2/1.6 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/1.6 topology video disable
set protocols ospf area 0.0.0.0 interface fe-1/2/1.6 topology voice
set protocols ospf area 0.0.0.0 interface fe-1/2/2.9 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/2.9 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/2.9 topology video
set protocols ospf area 0.0.0.0 interface lo0.93 passive
set policy-options policy-statement nhs then next-hop self
set routing-options autonomous-system 100
set routing-options topologies family inet topology voice
set routing-options topologies family inet topology video
set firewall family inet filter ef_path term ef from forwarding-class expedited-forwarding
set firewall family inet filter ef_path term ef then topology voice
set firewall family inet filter ef_path term video from source-address 11.19.132.0/24
set firewall family inet filter ef_path term video from source-address 11.19.133.0/24
set firewall family inet filter ef_path term video from source-address 11.19.142.0/24
set firewall family inet filter ef_path term video from source-address 11.19.144.0/24
set firewall family inet filter ef_path term video then topology video
set firewall family inet filter ef_path term default then accept
set class-of-service interfaces fe-1/2/0 unit 2 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/1 unit 6 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/2 unit 9 classifiers inet-precedence default
```

Device PE2

```
set interfaces fe-1/2/0 unit 38 description to-P2
set interfaces fe-1/2/0 unit 38 family inet filter input ef_path
set interfaces fe-1/2/0 unit 38 family inet address 10.0.0.38/30
set interfaces fe-1/2/1 unit 42 description to-P4
```



```

set interfaces fe-1/2/1 unit 42 family inet filter input ef_path
set interfaces fe-1/2/1 unit 42 family inet address 10.0.0.42/30
set interfaces fe-1/2/2 unit 45 description to-CE2
set interfaces fe-1/2/2 unit 45 family inet filter input ef_path
set interfaces fe-1/2/2 unit 45 family inet address 10.0.0.45/30
set interfaces lo0 unit 203 family inet address 10.255.165.203/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.203
set protocols bgp group ibgp family inet unicast topology voice community target:40:40
set protocols bgp group ibgp family inet unicast topology video community target:50:50
set protocols bgp group ibgp export nhs
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.93
set protocols bgp group ibgp neighbor 10.255.165.113
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols bgp group ebgp type external
set protocols bgp group ebgp local-address 10.0.0.45
set protocols bgp group ebgp family inet unicast topology voice community target:40:40
set protocols bgp group ebgp family inet unicast topology video community target:50:50
set protocols bgp group ebgp peer-as 102
set protocols bgp group ebgp neighbor 10.0.0.46
set protocols ospf topology voice topology-id 126
set protocols ospf topology video topology-id 52
set protocols ospf area 0.0.0.0 interface fe-1/2/0.38 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/0.38 topology video metric 200
set protocols ospf area 0.0.0.0 interface fe-1/2/0.38 topology voice
set protocols ospf area 0.0.0.0 interface fe-1/2/1.42 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/1.42 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/1.42 topology video
set protocols ospf area 0.0.0.0 interface lo0.203 passive
set policy-options policy-statement nhs then next-hop self
set routing-options autonomous-system 100
set routing-options topologies family inet topology voice
set routing-options topologies family inet topology video
set firewall family inet filter ef_path term ef from forwarding-class expedited-forwarding
set firewall family inet filter ef_path term ef then topology voice
set firewall family inet filter ef_path term video from source-address 11.19.132.0/24
set firewall family inet filter ef_path term video from source-address 11.19.133.0/24
set firewall family inet filter ef_path term video from source-address 11.19.142.0/24
set firewall family inet filter ef_path term video from source-address 11.19.144.0/24
set firewall family inet filter ef_path term video then topology video
set firewall family inet filter ef_path term default then accept
set class-of-service interfaces fe-1/2/0 unit 38 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/1 unit 42 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/2 unit 45 classifiers inet-precedence default

```

Device P1

```

set interfaces fe-1/2/0 unit 5 description to-PE1
set interfaces fe-1/2/0 unit 5 family inet filter input ef_path
set interfaces fe-1/2/0 unit 5 family inet address 10.0.0.5/30
set interfaces fe-1/2/1 unit 13 description to-P3
set interfaces fe-1/2/1 unit 13 family inet filter input ef_path
set interfaces fe-1/2/1 unit 13 family inet address 10.0.0.13/30
set interfaces fe-1/2/2 unit 17 description to-P4
set interfaces fe-1/2/2 unit 17 family inet filter input ef_path
set interfaces fe-1/2/2 unit 17 family inet address 10.0.0.17/30

```

```

set interfaces fe-1/2/3 unit 33 description to-P2
set interfaces fe-1/2/3 unit 33 family inet filter input ef_path
set interfaces fe-1/2/3 unit 33 family inet address 10.0.0.33/30
set interfaces lo0 unit 99 family inet address 10.255.165.99/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.99
set protocols bgp group ibgp family inet unicast topology voice community target:40:40
set protocols bgp group ibgp family inet unicast topology video community target:50:50
set protocols bgp group ibgp neighbor 10.255.165.93
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.113
set protocols ospf topology voice topology-id 126
set protocols ospf topology video topology-id 52
set protocols ospf area 0.0.0.0 interface fe-1/2/3.33 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/3.33 topology voice
set protocols ospf area 0.0.0.0 interface fe-1/2/3.33 topology video disable
set protocols ospf area 0.0.0.0 interface fe-1/2/2.17 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/2.17 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/2.17 topology video disable
set protocols ospf area 0.0.0.0 interface fe-1/2/1.13 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/1.13 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/1.13 topology video disable
set protocols ospf area 0.0.0.0 interface fe-1/2/0.5 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/0.5 topology voice
set protocols ospf area 0.0.0.0 interface fe-1/2/0.5 topology video disable
set protocols ospf area 0.0.0.0 interface lo0.99 passive
set routing-options autonomous-system 100
set routing-options topologies family inet topology voice
set routing-options topologies family inet topology video
set firewall family inet filter ef_path term ef from forwarding-class expedited-forwarding
set firewall family inet filter ef_path term ef then topology voice
set firewall family inet filter ef_path term video from source-address 11.19.132.0/24
set firewall family inet filter ef_path term video from source-address 11.19.133.0/24
set firewall family inet filter ef_path term video from source-address 11.19.142.0/24
set firewall family inet filter ef_path term video from source-address 11.19.144.0/24
set firewall family inet filter ef_path term video then topology video
set firewall family inet filter ef_path term default then accept
set class-of-service interfaces fe-1/2/0 unit 5 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/1 unit 13 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/2 unit 17 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/3 unit 33 classifiers inet-precedence default

```

Device P2

```

set interfaces fe-1/2/0 unit 22 description to-P3
set interfaces fe-1/2/0 unit 22 family inet filter input ef_path
set interfaces fe-1/2/0 unit 22 family inet address 10.0.0.22/30
set interfaces fe-1/2/1 unit 25 description to-P4
set interfaces fe-1/2/1 unit 25 family inet filter input ef_path
set interfaces fe-1/2/1 unit 25 family inet address 10.0.0.25/30
set interfaces fe-1/2/2 unit 34 description to-P1
set interfaces fe-1/2/2 unit 34 family inet filter input ef_path
set interfaces fe-1/2/2 unit 34 family inet address 10.0.0.34/30
set interfaces fe-1/2/3 unit 37 description to-PE2
set interfaces fe-1/2/3 unit 37 family inet filter input ef_path
set interfaces fe-1/2/3 unit 37 family inet address 10.0.0.37/30

```

```

set interfaces lo0 unit 113 family inet address 10.255.165.113/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.113
set protocols bgp group ibgp family inet unicast topology voice community target:40:40
set protocols bgp group ibgp family inet unicast topology video community target:50:50
set protocols bgp group ibgp neighbor 10.255.165.93
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols ospf topology voice topology-id 126
set protocols ospf topology video topology-id 52
set protocols ospf area 0.0.0.0 interface fe-1/2/2.34 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/2.34 topology voice
set protocols ospf area 0.0.0.0 interface fe-1/2/2.34 topology video disable
set protocols ospf area 0.0.0.0 interface fe-1/2/0.22 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/0.22 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/0.22 topology video metric 20
set protocols ospf area 0.0.0.0 interface fe-1/2/1.25 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/1.25 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/1.25 topology video disable
set protocols ospf area 0.0.0.0 interface fe-1/2/3.37 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/3.37 topology voice
set protocols ospf area 0.0.0.0 interface fe-1/2/3.37 topology video metric 200
set protocols ospf area 0.0.0.0 interface lo0.113 passive
set routing-options autonomous-system 100
set routing-options topologies family inet topology voice
set routing-options topologies family inet topology video
set firewall family inet filter ef_path term ef from forwarding-class expedited-forwarding
set firewall family inet filter ef_path term ef then topology voice
set firewall family inet filter ef_path term video from source-address 11.19.132.0/24
set firewall family inet filter ef_path term video from source-address 11.19.133.0/24
set firewall family inet filter ef_path term video from source-address 11.19.142.0/24
set firewall family inet filter ef_path term video from source-address 11.19.144.0/24
set firewall family inet filter ef_path term video then topology video
set firewall family inet filter ef_path term default then accept
set class-of-service interfaces fe-1/2/0 unit 22 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/1 unit 25 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/2 unit 34 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/3 unit 37 classifiers inet-precedence default

```

Device P3

```

set interfaces fe-1/2/0 unit 10 description to-PE1
set interfaces fe-1/2/0 unit 10 family inet filter input ef_path
set interfaces fe-1/2/0 unit 10 family inet address 10.0.0.10/30
set interfaces fe-1/2/1 unit 14 description to-P1
set interfaces fe-1/2/1 unit 14 family inet filter input ef_path
set interfaces fe-1/2/1 unit 14 family inet address 10.0.0.14/30
set interfaces fe-1/2/2 unit 21 description to-P2
set interfaces fe-1/2/2 unit 21 family inet filter input ef_path
set interfaces fe-1/2/2 unit 21 family inet address 10.0.0.21/30
set interfaces fe-1/2/3 unit 29 description to-P4
set interfaces fe-1/2/3 unit 29 family inet filter input ef_path
set interfaces fe-1/2/3 unit 29 family inet address 10.0.0.29/30
set interfaces lo0 unit 111 family inet address 10.255.165.111/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.111

```

```

set protocols bgp group ibgp family inet unicast topology voice community target:40:40
set protocols bgp group ibgp family inet unicast topology video community target:50:50
set protocols bgp group ibgp neighbor 10.255.165.93
set protocols bgp group ibgp neighbor 10.255.165.113
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols ospf topology voice topology-id 126
set protocols ospf topology video topology-id 52
set protocols ospf area 0.0.0.0 interface fe-1/2/3.29 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/3.29 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/3.29 topology video
set protocols ospf area 0.0.0.0 interface fe-1/2/2.21 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/2.21 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/2.21 topology video metric 20
set protocols ospf area 0.0.0.0 interface fe-1/2/1.14 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/1.14 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/1.14 topology video disable
set protocols ospf area 0.0.0.0 interface fe-1/2/0.10 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/0.10 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/0.10 topology video
set protocols ospf area 0.0.0.0 interface lo0.111 passive
set routing-options autonomous-system 100
set routing-options topologies family inet topology voice
set routing-options topologies family inet topology video
set firewall family inet filter ef_path term ef from forwarding-class expedited-forwarding
set firewall family inet filter ef_path term ef then topology voice
set firewall family inet filter ef_path term video from source-address 11.19.132.0/24
set firewall family inet filter ef_path term video from source-address 11.19.133.0/24
set firewall family inet filter ef_path term video from source-address 11.19.142.0/24
set firewall family inet filter ef_path term video from source-address 11.19.144.0/24
set firewall family inet filter ef_path term video then topology video
set firewall family inet filter ef_path term default then accept
set class-of-service interfaces fe-1/2/0 unit 10 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/1 unit 14 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/2 unit 21 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/3 unit 29 classifiers inet-precedence default

```

Device P4

```

set interfaces fe-1/2/0 unit 18 description to-P1
set interfaces fe-1/2/0 unit 18 family inet filter input ef_path
set interfaces fe-1/2/0 unit 18 family inet address 10.0.0.18/30
set interfaces fe-1/2/1 unit 26 description to-P2
set interfaces fe-1/2/1 unit 26 family inet filter input ef_path
set interfaces fe-1/2/1 unit 26 family inet address 10.0.0.26/30
set interfaces fe-1/2/2 unit 30 description to-P3
set interfaces fe-1/2/2 unit 30 family inet filter input ef_path
set interfaces fe-1/2/2 unit 30 family inet address 10.0.0.30/30
set interfaces fe-1/2/3 unit 41 description to-PE2
set interfaces fe-1/2/3 unit 41 family inet filter input ef_path
set interfaces fe-1/2/3 unit 41 family inet address 10.0.0.41/30
set interfaces lo0 unit 95 family inet address 10.255.165.95/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.95
set protocols bgp group ibgp family inet unicast topology voice community target:40:40
set protocols bgp group ibgp family inet unicast topology video community target:50:50
set protocols bgp group ibgp neighbor 10.255.165.93

```

```

set protocols bgp group ibgp neighbor 10.255.165.113
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols ospf topology voice topology-id 126
set protocols ospf topology video topology-id 52
set protocols ospf area 0.0.0.0 interface fe-1/2/2.30 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/2.30 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/2.30 topology video
set protocols ospf area 0.0.0.0 interface fe-1/2/0.18 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/0.18 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/0.18 topology video disable
set protocols ospf area 0.0.0.0 interface fe-1/2/0.18 topology video metric 20
set protocols ospf area 0.0.0.0 interface fe-1/2/1.26 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/1.26 topology voice disable
set protocols ospf area 0.0.0.0 interface fe-1/2/1.26 topology video disable
set protocols ospf area 0.0.0.0 interface fe-1/2/3.41 metric 10
set protocols ospf area 0.0.0.0 interface fe-1/2/3.41 topology voice
set protocols ospf area 0.0.0.0 interface fe-1/2/3.41 topology video
set protocols ospf area 0.0.0.0 interface lo0.95 passive
set routing-options autonomous-system 100
set routing-options topologies family inet topology voice
set routing-options topologies family inet topology video
set firewall family inet filter ef_path term ef from forwarding-class expedited-forwarding
set firewall family inet filter ef_path term video from source-address 11.19.132.0/24
set firewall family inet filter ef_path term video from source-address 11.19.133.0/24
set firewall family inet filter ef_path term video from source-address 11.19.142.0/24
set firewall family inet filter ef_path term video from source-address 11.19.144.0/24
set firewall family inet filter ef_path term video then topology video
set firewall family inet filter ef_path term default then accept
set class-of-service interfaces fe-1/2/0 unit 18 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/1 unit 26 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/2 unit 30 classifiers inet-precedence default
set class-of-service interfaces fe-1/2/3 unit 41 classifiers inet-precedence default

```

Configuring Device CE1

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 CE1:

1. Configure the interfaces.

For demonstration purposes, the example places an Ethernet interface into loopback mode and configures several addresses on this loopback interface. The addresses are then announced to the network.

```

[edit interfaces]
user@CE1# set fe-0/1/0 fastether-options loopback
user@CE1# set fe-0/1/0 unit 0 family inet address 11.19.130.1/24
user@CE1# set fe-0/1/0 unit 0 family inet address 11.19.131.1/24
user@CE1# set fe-0/1/0 unit 0 family inet address 11.19.132.1/24

```

```
user@CE1# set fe-1/2/0 unit 1 description to-PE1
user@CE1# set fe-1/2/0 unit 1 family inet address 10.0.0.1/30
```

```
user@CE1# set lo0 unit 97 family inet address 10.255.165.97/32 primary
```

2. Configure the external BGP (EBGP) connection to Device PE1.

```
[edit protocols bgp group ebgp]
user@CE1# set type external
user@CE1# set local-address 10.0.0.1
user@CE1# set peer-as 100
user@CE1# set neighbor 10.0.0.2
```

3. Configure the routing policy that announces the addresses that are configured on interface fe-0/1/0.

```
[edit policy-options policy-statement inject_directs]
user@CE1# set term a from protocol direct
user@CE1# set term a from interface fe-0/1/0.0
user@CE1# set term a then next policy
user@CE1# set term a then accept
user@CE1# set term b then reject
```

4. Configure the routing policy that tags voice routes with the video community attribute, and video routes with the voice community attribute.

```
[edit policy-options policy-statement set_community]
user@CE1# set term a from route-filter 11.19.130.0/24 exact
user@CE1# set term a from route-filter 11.19.131.0/24 exact
user@CE1# set term a then community add voice
user@CE1# set term a then accept
```

```
user@CE1# set term b from route-filter 11.19.132.0/24 exact
user@CE1# set term b from route-filter 11.19.133.0/24 exact
user@CE1# set term b then community add video
user@CE1# set term b then accept
```

```
user@CE1# set term default then accept
```

```
[edit policy-options community]
user@CE1# set video members target:50:50
user@CE1# set voice members target:40:40
```

5. Apply the **set_community** export policy so that direct routes are exported from the routing table into BGP.

Apply the **inject_directs** export policy to announce the addresses that are configured on interface fe-0/1/0.

```
[edit protocols bgp group ebgp]
user@CE1# set export set_community
user@CE1# set export inject_directs
```

6. Configure the autonomous system (AS) number.

```
[edit routing-options]
user@CE1# set autonomous-system 101
```

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@CE1# show interfaces
fe-0/1/0 {
  fastether-options {
    loopback;
  }
  unit 0 {
    family inet {
      address 11.19.130.1/24;
      address 11.19.131.1/24;
      address 11.19.132.1/24;
    }
  }
}
fe-1/2/0 {
  unit 1 {
    description to-PE1;
    family inet {
      address 10.0.0.1/30;
    }
  }
}
lo0 {
  unit 97 {
    family inet {
      address 10.255.165.97/32 {
        primary;
      }
    }
  }
}
}

user@CE1# show protocols
bgp {
  group ebgp {
    type external;
    local-address 10.0.0.1;
    export [ set_community inject_directs ];
    peer-as 100;
    neighbor 10.0.0.2;
  }
}

user@CE1# show policy-options
policy-statement inject_directs {
  term a {
    from {
      protocol direct;
      interface fe-0/1/0.0;
    }
    then {
      next policy;
    }
  }
}

```

```
        accept;
    }
}
term b {
    then reject;
}
}
policy-statement set_community {
    term a {
        from {
            route-filter 11.19.130.0/24 exact;
            route-filter 11.19.131.0/24 exact;
        }
        then {
            community add voice;
            accept;
        }
    }
    term b {
        from {
            route-filter 11.19.132.0/24 exact;
            route-filter 11.19.133.0/24 exact;
        }
        then {
            community add video;
            accept;
        }
    }
    term default {
        then accept;
    }
}
community video members target:50:50;
community voice members target:40:40;

user@CE1# show routing-options
autonomous-system 101;
```

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

Configuring Device PE1

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 PE1:

1. Configure the interfaces.

The forwarding plane uses a firewall filter to indicate which topology forwarding table traffic should use. In this case, you must configure a firewall filter on all interfaces related to routing topologies. In general, all multitopology OSPF interfaces in the core where topologies are configured have input firewall filters. In addition, the ingress interfaces, where traffic from a CE device enters a PE device toward the core, have firewall filters configured. This configuration on Device PE1 shows a

firewall filter applied to the ingress interface (connected to the CE device) and to the two core-facing interfaces (connected to Device P1 and Device P3).

```
[edit interfaces]
```

```
user@PE1# set fe-1/2/0 unit 2 description to-CE1
user@PE1# set fe-1/2/0 unit 2 family inet filter input ef_path
user@PE1# set fe-1/2/0 unit 2 family inet address 10.0.0.2/30
```

```
user@PE1# set fe-1/2/1 unit 6 description to-P1
user@PE1# set fe-1/2/1 unit 6 family inet filter input ef_path
user@PE1# set fe-1/2/1 unit 6 family inet address 10.0.0.6/30
```

```
user@PE1# set fe-1/2/2 unit 9 description to-P3
user@PE1# set fe-1/2/2 unit 9 family inet filter input ef_path
user@PE1# set fe-1/2/2 unit 9 family inet address 10.0.0.9/30
```

```
user@PE1# set lo0 unit 93 family inet address 10.255.165.93/32 primary
```

2. Configure the autonomous system (AS) number.

```
[edit routing-options]
```

```
user@PE1# set autonomous-system 100
```

3. Configure BGP.

```
[edit protocols bgp group ibgp]
```

```
user@PE1# set type internal
user@PE1# set local-address 10.255.165.93
user@PE1# set neighbor 10.255.165.111
user@PE1# set neighbor 10.255.165.203
user@PE1# set neighbor 10.255.165.113
user@PE1# set neighbor 10.255.165.95
user@PE1# set neighbor 10.255.165.99
```

```
[edit protocols bgp group ebgp]
```

```
user@PE1# set type external
user@PE1# set local-address 10.0.0.2
user@PE1# set peer-as 101
user@PE1# set neighbor 10.0.0.1
```

4. Configure a next-hop self routing policy to ensure that the IBGP devices use the loopback address on Device PE1 as the next-hop address on all IBGP route advertisements.

This way, Device PE1 serves as the gateway router for EBGp routes.

```
[edit policy-options policy-statement nhs]
```

```
user@PE1# set then next-hop self
```

5. Apply the next-hop self policy to the IBGP sessions.

```
[edit protocols bgp group ibgp]
```

```
user@PE1# set export nhs
```

6. Configure the voice and video topologies, which enable you to use these topologies with OSPF and BGP.

The names **voice** and **video** are local to the router. The names are not propagated beyond this router. However, for management purposes, a consistent naming scheme across routers in a multitopology environment is convenient.

```
[edit routing-options topologies family inet]
user@PE1# set topology voice
user@PE1# set topology video
```

7. Apply the community tags to identify the voice and video topologies by configuring a routing topology name and BGP community value.

In Junos OS, multitopology support for BGP is based on the community value in a BGP route. This configuration determines the association between a topology and one or more community values and populates the topology routing tables. Arriving BGP updates that have a matching community value are replicated in the associated topology routing table. You decide which BGP community values are associated with a given topology.

This configuration causes BGP updates that are received with community value **target:40:40** to be added into topology routing table **:voice.inet.0** (in addition to the default routing table **inet.0**). Updates that are received with community value **target:50:50** are added into topology routing table **:video.inet.0** (in addition to the default routing table **inet.0**).

```
[edit protocols bgp group ibgp family inet unicast]
user@PE1# set topology voice community target:40:40
user@PE1# set topology video community target:50:50
```

```
[edit protocols bgp group ebgp family inet unicast]
user@PE1# set topology voice community target:40:40
user@PE1# set topology video community target:50:50
```

8. Enable and disable multitopology OSPF on particular interfaces.

Enable multitopology OSPF designations only on desired interfaces, as shown in [Figure 5 on page 16](#). On interface fe-1/2/1.6 facing Device P1, enable the voice topology, and disable the video topology. On interface fe-1/2/2.9 facing Device P3, enable the video topology, and disable the voice topology.

When a topology ID is configured under OSPF, the topology is automatically enabled on all interfaces under OSPF. To disable a topology or to add a metric, you must add an explicit configuration.

For readability purposes, each topology is configured under each desired OSPF interface even though this default behavior occurs when the topology ID is configured. Configure higher metric values on a link to make the link less preferred than another available link.

```
[edit protocols ospf ]
user@PE1# set topology voice topology-id 126
user@PE1# set topology video topology-id 52
```

```
[edit protocols ospf area 0.0.0.0]
user@PE1# set interface fe-1/2/1.6 metric 10
user@PE1# set interface fe-1/2/1.6 topology video disable
user@PE1# set interface fe-1/2/1.6 topology voice
```

```

user@PE1# set interface fe-1/2/2.9 metric 10
user@PE1# set interface fe-1/2/2.9 topology voice disable
user@PE1# set interface fe-1/2/2.9 topology video

```

```

user@PE1# set interface lo0.93 passive

```

9. Configure the firewall filter.

After routing topologies are configured, traffic must go through a firewall filter to make use of routing topology forwarding tables. For basic routing topologies, where traffic is first entering the core network, apply an input firewall filter to the ingress interface. Additionally, add firewall filters to interfaces where multitopology OSPF is configured. All routers must use the same firewall filter to associate packets with a topology to ensure consistent forwarding and to avoid routing loops or packet loss.

The forwarding plane handles traffic as it enters the router and exits out a particular interface. To inspect traffic and use a specified topology forwarding table to perform next-hop lookups, configure an input firewall filter on each interface where routing topology support is desired. Use a regular firewall filter to identify packet characteristics.

In general, for application-level differentiation, it is convenient to use DiffServ code points (DSCPs). When there is a firewall filter match, the firewall instructs the route lookup to use a particular topology forwarding table. Packet attributes are identified in the **from** clause, followed by a **then** clause indicating the topology forwarding table for use in forwarding next-hop lookups. This configuration notifies the router which traffic uses a routing topology forwarding table and which traffic uses the default forwarding table. The last term, which is named **default**, specifies the use of the default forwarding table.

These firewall configurations show source addresses and DSCPs used to sort voice, video, and default traffic. DSCPs are practical because you can set them at or near a CE device and because the information is intact across the network. For instance, here class of service (CoS) is configured for expedited traffic. DSCPs are also practical when the same IP address is used for different applications.

```

[edit firewall family inet filter ef_path]
user@PE1# set term ef from forwarding-class expedited-forwarding
user@PE1# set term ef then topology voice

```

```

user@PE1# set term video from source-address 11.19.132.0/24
user@PE1# set term video from source-address 11.19.133.0/24
user@PE1# set term video from source-address 11.19.142.0/24
user@PE1# set term video from source-address 11.19.144.0/24
user@PE1# set term video then topology video

```

```

user@PE1# set term default then accept

```

10. Enable CoS on the interfaces.

```

[edit class-of-service interfaces]
user@PE1# set fe-1/2/0 unit 2 classifiers inet-precedence default
user@PE1# set fe-1/2/1 unit 6 classifiers inet-precedence default

```

```
user@PE1# set fe-1/2/2 unit 9 classifiers inet-precedence default
```

Results From configuration mode, confirm your configuration by entering the **show interfaces**, **show protocols**, **show policy-options**, **show routing-options**, **show firewall**, and **show class-of-service** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@PE1# show interfaces
fe-1/2/0 {
  unit 2 {
    description to-CE1;
    family inet {
      filter {
        input ef_path;
      }
      address 10.0.0.2/30;
    }
  }
}
fe-1/2/1 {
  unit 6 {
    description to-P1;
    family inet {
      filter {
        input ef_path;
      }
      address 10.0.0.6/30;
    }
  }
}
fe-1/2/2 {
  unit 9 {
    description to-P3;
    family inet {
      filter {
        input ef_path;
      }
      address 10.0.0.9/30;
    }
  }
}
lo0 {
  unit 93 {
    family inet {
      address 10.255.165.93/32 {
        primary;
      }
    }
  }
}

user@PE1# show protocols
bgp {
  group ibgp {
    type internal;
```

```

local-address 10.255.165.93;
family inet {
    unicast {
        topology voice {
            community target:40:40;
        }
        topology video {
            community target:50:50;
        }
    }
}
export nhs;
neighbor 10.255.165.111;
neighbor 10.255.165.203;
neighbor 10.255.165.113;
neighbor 10.255.165.95;
neighbor 10.255.165.99;
}
group ebgp {
    type external;
    local-address 10.0.0.2;
    family inet {
        unicast {
            topology voice {
                community target:40:40;
            }
            topology video {
                community target:50:50;
            }
        }
    }
    peer-as 101;
    neighbor 10.0.0.1;
}
}
ospf {
    topology voice topology-id 126;
    topology video topology-id 52;
    area 0.0.0.0 {
        interface fe-1/2/1.6 {
            metric 10;
            topology video disable;
            topology voice;
        }
        interface fe-1/2/2.9 {
            metric 10;
            topology voice disable;
            topology video;
        }
        interface lo0.93 {
            passive;
        }
    }
}
}
user@PE1# show policy-options

```

```
policy-statement nhs {
  then {
    next-hop self;
  }
}

user@PE1# show routing-options
autonomous-system 100;
topologies {
  family inet {
    topology voice;
    topology video;
  }
}

user@PE1# show firewall
family inet {
  filter ef_path {
    term ef {
      from {
        forwarding-class expedited-forwarding;
      }
      then topology voice;
    }
    term video {
      from {
        source-address {
          11.19.132.0/24;
          11.19.133.0/24;
          11.19.142.0/24;
          11.19.144.0/24;
        }
      }
      then topology video;
    }
    term default {
      then accept;
    }
  }
}

user@PE1# show class-of-service
interfaces {
  fe-1/2/0 {
    unit 2 {
      classifiers {
        inet-precedence default;
      }
    }
    unit 6 {
      classifiers {
        inet-precedence default;
      }
    }
    unit 9 {
      classifiers {
        inet-precedence default;
      }
    }
  }
}
```

```

    }
  }
}

```

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

Verification

Confirm that the configuration is working properly.

- [Verifying the OSPF Interfaces on page 33](#)
- [Verifying the Routes on page 34](#)
- [Checking the Resolving BGP Next Hops on page 35](#)
- [Examining the Protocol Next Hop on page 36](#)
- [Verifying the OSPF Neighbor on page 36](#)
- [Checking the Router LSA on page 37](#)
- [Checking How Traffic Traverses the Network on page 38](#)

Verifying the OSPF Interfaces

Purpose Verify that the OSPF interfaces are configured to belong to one or more topologies.

Action From operational mode, enter the **show (ospf | ospf3) interface interface-name detail** command.

```

user@PE1> show ospf interface fe-1/2/1.6 detail

```

| Interface | State | Area | DR ID | BDR ID | Nbrs |
|------------|-------|---------|---------------|---------------|------|
| fe-1/2/1.6 | DR | 0.0.0.0 | 10.255.165.93 | 10.255.165.99 | 1 |

```

Type: LAN, Address: 10.0.0.6, Mask: 255.255.255.252, MTU: 1500, Cost: 10
DR addr: 10.0.0.6, BDR addr: 10.0.0.5, Priority: 128
Adj count: 1
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Protection type: None
Topology default (ID 0) -> Cost: 10
Topology video (ID 52) -> Disabled, Cost: 10
Topology voice (ID 126) -> Cost: 10

```

Meaning This output shows that the voice topology was added to the fe-1/2/1.6 interface on Device PE1. The topology name is voice, and the MT-ID is 126. The video topology is disabled on this interface. The cost of the interface is 10.

The Router-LSA originated and flooded by the router includes all relevant topology information for specific interfaces, such as MT-ID and metric. If MTR is not configured on an OSPF interface, then the Router-LSA does not include any topology information for that interface. OSPF neighbors might or might not support multitopology OSPF. That is, a particular link is not used to calculate OSPF routes for a topology unless routers at both ends of the link announce that link as part of the topology. If multitopology OSPF is not supported in neighboring OSPF routers or is not configured to do so, then topology information in LSAs received by the neighbor is ignored.

Verifying the Routes

Purpose Make sure that the routes are in the expected routing tables and that the expected communities are attached to the routes.

Action From operational mode, enter the `show route detail` command on Device PE1.

```
user@PE1> show route 11.19.130.0/24 detail

inet.0: 29 destinations, 30 routes (29 active, 0 holddown, 0 hidden)
11.19.130.0/24 (1 entry, 1 announced)
  *BGP   Preference: 170/-101
        Next hop type: Router, Next hop index: 812
        Address: 0xb9f064c
        Next-hop reference count: 22
        Source: 10.0.0.1
        Next hop: 10.0.0.1 via fe-1/2/0.2, selected
        Session Id: 0x600004
        State: <Active Ext>
        Local AS: 100 Peer AS: 101
        Age: 3d 21:44:07
        Task: BGP_101.10.0.0.1+51873
        Announcement bits (3): 0-KRT 3-BGP_RT_Background 4-Resolve tree
3
        AS path: 101 I
        Communities: target:40:40
        Accepted
        Localpref: 100
        Router ID: 10.255.165.97
        Secondary Tables: :voice.inet.0

:voice.inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
11.19.130.0/24 (1 entry, 1 announced)
  *BGP   Preference: 170/-101
        Next hop type: Router, Next hop index: 812
        Address: 0xb9f064c
        Next-hop reference count: 22
        Source: 10.0.0.1
        Next hop: 10.0.0.1 via fe-1/2/0.2, selected
        Session Id: 0x600004
        State: <Secondary Active IndepResolution Ext>
        Local AS: 100 Peer AS: 101
        Age: 3d 21:44:07
        Task: BGP_101.10.0.0.1+51873
        Announcement bits (2): 0-KRT 1-Resolve tree 1
        AS path: 101 I
        Communities: target:40:40
        Accepted
        Localpref: 100
        Router ID: 10.255.165.97
        Primary Routing Table inet.0
```

Meaning This output shows BGP route 11.19.130.0/24 with community value target:40:40. Because the route matches the criteria for the voice topology, it is added to both the default and voice topology routing tables (`inet.0` and `:voice.inet.0`). Device PE1 learns the route from Device CE1 through EBGP and then injects the route into IBGP.

Checking the Resolving BGP Next Hops

Purpose Check the protocol next hop and forwarding next hop.

Action From operational mode, enter the `show route detail` command on Device PE2.

```

user@PE2> show route 11.19.130.0/24 detail
inet.0: 29 destinations, 30 routes (29 active, 0 holddown, 0 hidden)
11.19.130.0/24 (1 entry, 1 announced)
  *BGP      Preference: 170/-101
            Next hop type: Indirect
            Address: 0xb9f0e04
            Next-hop reference count: 12
            Source: 10.255.165.93
            Next hop type: Router, Next hop index: 262153
            Next hop: 10.0.0.37 via fe-1/2/0.38
            Session Id: 0x700004
            Next hop: 10.0.0.41 via fe-1/2/1.42, selected
            Session Id: 0x700005
            Protocol next hop: 10.255.165.93
            Indirect next hop: bb8c000 262154 INH Session ID: 0x700007
            State: <Active Int Ext>
            Local AS: 100 Peer AS: 100
            Age: 3d 4:27:40 Metric2: 30
            Task: BGP_100.10.255.165.93+179
            Announcement bits (3): 0-KRT 3-BGP_RT_Background 4-Resolve tree
3
            AS path: 101 I
            Communities: target:40:40
            Accepted
            Localpref: 100
            Router ID: 10.255.165.93
            Secondary Tables: :voice.inet.0

:voice.inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
11.19.130.0/24 (1 entry, 1 announced)
  *BGP      Preference: 170/-101
            Next hop type: Indirect
            Address: 0xb9f0f34
            Next-hop reference count: 6
            Source: 10.255.165.93
            Next hop type: Router, Next hop index: 1188
            Next hop: 10.0.0.37 via fe-1/2/0.38, selected
            Session Id: 0x700004
            Protocol next hop: 10.255.165.93
            Indirect next hop: bb8c1d8 262177 INH Session ID: 0x700007
            State: <Secondary Active IndepResolution Int Ext>
            Local AS: 100 Peer AS: 100
            Age: 3d 2:00:20 Metric2: 30
            Task: BGP_100.10.255.165.93+179
            Announcement bits (2): 0-KRT 1-Resolve tree 1
            AS path: 101 I
            Communities: target:40:40
            Accepted
            Localpref: 100
            Router ID: 10.255.165.93
            Primary Routing Table inet.0

```

Meaning A typical IBGP core has BGP routes with protocol next hops that resolve using the underlying IGP routes. IBGP routes in a topology routing table have protocol next-hop IP addresses. By default, the same topology routing table is used to look up and resolve the protocol next-hop IP address to a forwarding next hop. This output from Device PE2 shows the same BGP route as seen in the previous example: 11.19.130.0/24. The route is being shown from a different perspective, that is, from Device PE2 as an IBGP route. Similarly, this IBGP route is added to both `inet.0` and `:voice.inet.0` on Device PE2. However, while each route has the same protocol next hop, each route has a different forwarding next hop (ge-0/0/3.0 instead of ge-0/1/4.0). The reason for this difference is when the protocol next-hop IP address 10.255.165.93 is resolved, it uses the corresponding routing table (`inet.0` or `:voice.inet.0`) to look up the protocol next hop.

Examining the Protocol Next Hop

Purpose Check the protocol next hop and forwarding next hop.

Action From operational mode, enter the `show route` command on Device PE2.

```
user@PE2> show route 10.255.165.93
inet.0: 29 destinations, 30 routes (29 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.165.93/32    *[OSPF/10] 3d 04:37:26, metric 30
                  > to 10.0.0.37 via fe-1/2/0.38
                  to 10.0.0.41 via fe-1/2/1.42

:voice.inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.165.93/32    *[OSPF/10] 3d 02:10:04, metric 30
                  > to 10.0.0.37 via fe-1/2/0.38

:video.inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.165.93/32    *[OSPF/10] 3d 02:03:16, metric 30
                  > to 10.0.0.41 via fe-1/2/1.42
```

Meaning This output from Device PE2 shows the protocol next hop of 11.19.130.0/24, which is IP address 10.255.165.93, thus further demonstrating how IBGP route 11.19.130.0/24 resolves its protocol next hop. The forwarding next hops of 10.255.165.93 match the IBGP forwarding next hops of route 11.19.130.0/24 as shown in the previous example. Observe here that the IP address 10.255.165.93 is also in routing table `:video.inet.0`. This address is the loopback address of Device PE1, and as such, resides in all three routing tables. This example also shows how traffic entering Device PE2 destined to 11.19.130.0/24 exits out different interfaces depending on its associated topology. The actual traffic is marked in such a way that a firewall filter can direct the traffic to use a particular topology routing table.

Verifying the OSPF Neighbor

Purpose Make sure that the expected topologies are enabled on the OSPF neighbor.

Action From operational mode, enter the **show ospf neighbor 10.0.0.21 extensive** command on Device P2.

```
user@P2> show ospf neighbor 10.0.0.21 extensive
Address          Interface          State    ID                Pri  Dead
10.0.0.21        fe-1/2/0.22        Full    10.255.165.111   128   39
Area 0.0.0.0, opt 0x52, DR 10.0.0.22, BDR 10.0.0.21
Up 3d 06:09:50, adjacent 3d 06:09:50
Topology default (ID 0) -> Bidirectional
Topology video (ID 52) -> Bidirectional
```

Meaning This Device P2 output shows OSPF neighbor PE2 (10.0.0.21), where multitopology OSPF default and video are multitopology OSPF participants. The **Bidirectional** flag shows that the neighbor is configured using the same multitopology OSPF ID.

Checking the Router LSA

Purpose Check the links where video and voice topologies are enabled.

Action From operational mode, enter the **show ospf database lsa-id 10.255.165.203 extensive** command on Device P2.

```
user@P2> show ospf database lsa-id 10.255.165.203 extensive

OSPF database, Area 0.0.0.0
Type      ID                Adv Rtr          Seq          Age  Opt  Cksum  Len
Router    10.255.165.203    10.255.165.203   0x800000063   1552 0x22 0xdff3  80
bits 0x0, link count 3
id 10.255.165.203, data 255.255.255.255, Type Stub (3)
Topology count: 2, Default metric: 0
Topology video (ID 52) -> Metric: 0
Topology voice (ID 126) -> Metric: 0
id 10.0.0.38, data 10.0.0.38, Type Transit (2)
Topology count: 2, Default metric: 10
Topology video (ID 52) -> Metric: 200
Topology voice (ID 126) -> Metric: 10
id 10.0.0.42, data 10.0.0.42, Type Transit (2)
Topology count: 1, Default metric: 10
Topology video (ID 52) -> Metric: 10
Topology default (ID 0)
Type: Transit, Node ID: 10.0.0.42
Metric: 10, Bidirectional
Type: Transit, Node ID: 10.0.0.38
Metric: 10, Bidirectional
Topology video (ID 52)
Type: Transit, Node ID: 10.0.0.42
Metric: 10, Bidirectional
Type: Transit, Node ID: 10.0.0.38
Metric: 200, Bidirectional
Topology voice (ID 126)
Type: Transit, Node ID: 10.0.0.38
Metric: 10, Bidirectional
Aging timer 00:34:08
Installed 00:25:49 ago, expires in 00:34:08, sent 00:25:47 ago
Last changed 3d 01:45:51 ago, Change count: 10
```

Meaning This Device P2 output shows the Router-LSA originated by Device PE2. The LSA shows links where video and voice topologies are enabled (in addition to the default topology).

Checking How Traffic Traverses the Network

Purpose Make sure that the expected paths are used.

Action From operational mode, enter the `traceroute` command on Device CE1.

The first example output shows a traceroute over the voice topology goes from Device CE1 to Device CE2 where DSCPs are set. The routes are resolved over `:voice.inet.0`. This traceroute path follows the voice path CE1-PE1-P1-P2-PE2-CE2

```
user@CE1> traceroute 11.19.140.1 source 11.19.130.1 tos 160
```

```
traceroute to 11.19.140.1 (11.19.140.1) from 11.19.130.1, 30 hops max, 40 byte packets
```

```
 1 10.0.0.2 (10.0.0.2)  2.015 ms  1.924 ms  1.770 ms
 2 10.0.0.5 (10.0.0.5)  1.890 ms  1.010 ms  0.974 ms
 3 10.0.0.34 (10.0.0.34) 0.986 ms  1.031 ms  0.973 ms
 4 10.0.0.38 (10.0.0.38) 1.213 ms  1.065 ms  1.154 ms
 5 11.19.140.1 (11.19.140.1) 1.696 ms  4.286 ms  1.332 ms
```

This output shows a traceroute from Device CE1 to Device CE2 for voice where no DSCPs are set. The routes are resolved over `inet.0`, and the resulting path is different from the previous case where the DSCPs are set. This traceroute path follows the default path CE1-PE1-P4-PE2-CE2.

```
user@CE1> traceroute 11.19.140.1 source 11.19.130.1
```

```
traceroute to 11.19.140.1 (11.19.140.1) from 11.19.130.1, 30 hops max, 40 byte packets
```

```
 1 10.0.0.2 (10.0.0.2)  1.654 ms  1.710 ms  1.703 ms
 2 10.0.0.5 (10.0.0.5)  1.790 ms  1.045 ms  0.975 ms
 3 10.0.0.18 (10.0.0.18) 0.989 ms  1.041 ms  0.983 ms
 4 10.0.0.42 (10.0.0.42) 0.994 ms  1.036 ms  1.002 ms
 5 11.19.140.1 (11.19.140.1) 1.329 ms  2.248 ms  2.225 ms
```

This output shows a traceroute from Device CE1 to Device CE2 for video traffic where the firewall filter is based on the destination address. The routes are resolved over `:video.inet.0`. This traceroute follows the video path CE1-PE1-P3-P4-PE2-CE2.

```
user@CE1> traceroute 11.19.142.1 source 11.19.132.1
```

```
traceroute to 11.19.142.1 (11.19.142.1) from 11.19.132.1, 30 hops max, 40 byte packets
```

```
 1 10.0.0.2 (10.0.0.2)  1.126 ms  1.300 ms  0.995 ms
 2 10.0.0.10 (10.0.0.10) 0.981 ms  1.018 ms  0.991 ms
 3 10.0.0.30 (10.0.0.30) 0.997 ms  1.886 ms  1.952 ms
 4 10.0.0.42 (10.0.0.42) 1.800 ms  1.038 ms  0.980 ms
 5 11.19.142.1 (11.19.142.1) 1.367 ms  1.352 ms  1.328 ms
```

This output shows a traceroute from Device CE1 to Device CE2 for video where DSCPs are set. The DSCP bits are directing Device PE1 to use the topology table `:voice.inet.0`. Because there is no entry in the voice routing table for video routes, traffic is dropped.

```
user@CE1> traceroute 11.19.142.1 source 11.19.132.1 tos 160
```

```
traceroute to 11.19.142.1 (11.19.142.1) from 11.19.132.1, 30 hops max, 40 byte
```

packets

```
1 10.0.0.2 (10.0.0.2) 1.135 ms !N 1.007 ms !N 0.954 ms !N
```

Related Documentation

- [Example: Configuring Multitopology Routing Based on a Multicast Source on page 39](#)

Example: Configuring Multitopology Routing Based on a Multicast Source

- [Understanding Multitopology Routing Based on a Multicast Source on page 39](#)
- [Example: Configuring Multitopology Routing Based on a Multicast Source on page 41](#)

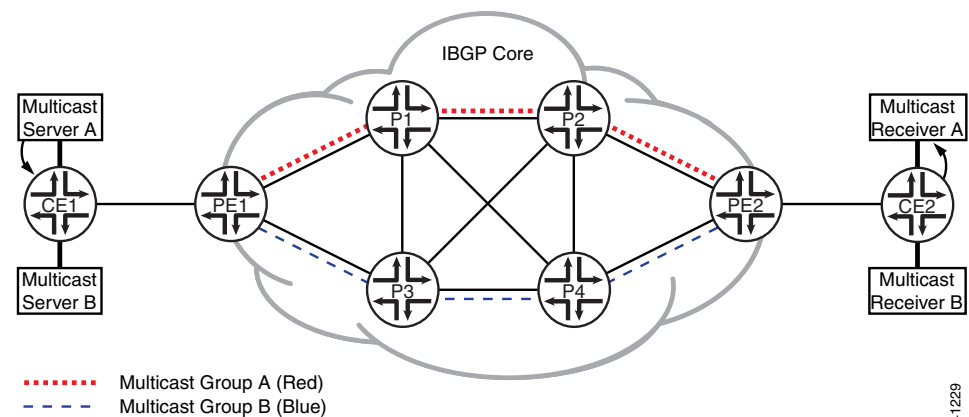
Understanding Multitopology Routing Based on a Multicast Source

Protocol Independent Multicast (PIM), in conjunction with multitopology routing extensions to OSPF (multitopology OSPF) and BGP, can direct multicast traffic over particular paths based on traffic characteristics.

Junos OS provides a mechanism whereby multicast traffic traverses user-specified topology paths based on the sender's source address. Multitopology routing (MTR) is used for OSPF, BGP, and route resolution over the specified topology routing tables. OSPF and BGP independently populate the routing table used by PIM. Firewall filters are not required because the multicast forwarding plane uses the multicast tree after it has been built.

[Figure 6 on page 39](#) shows a diagram of routing topology paths, where the dashed lines are associated with multicast group A (topology red), and the dotted lines are associated with multicast group B (topology blue).

Figure 6: Core Links Configured to Prefer Specified Routing Topologies



Two copies of the same stream enter Device PE1 and then traverse separate paths over the internal BGP (IBGP) core.

This solution leverages Junos OS features that allow particular routing tables to perform route resolution using specified routing tables.

The configuration includes a combination of the following features:

- BGP communities
- Separate IBGP next hops belonging to user-specified OSPF routing topologies
- Route resolution over user-specified topology routing tables
- A separate routing table (**inet.2**) for multicast protocols

Commonly, networks use a separate routing table for multicast. In Junos OS, the multicast routing table is **inet.2**. Routing topologies are grouped based on BGP communities. Each group represents a set of IP addresses associated with multicast servers and receivers. Primarily, the group must be related to the set of servers because the multicast receivers initiate tree creation toward these servers. Multicast traffic directed downstream toward receivers uses the previously created PIM tree, and therefore the forwarding plane does not need to know about routing topologies.

PIM uses the **inet.2** routing table for lookups of multicast source addresses. These IP addresses used for tree creation are IP unicast addresses. The customer edge (CE) routers, nearest to the multicast servers, announce the multicast source IP addresses to the provider edge (PE) routers using external BGP (EBGP). They are announced with both **family inet** unicast and **family inet multicast**, thus causing the BGP route to be added to the default routing table **inet.0** and to **inet.2**.

Both versions of the route are injected by the PE router into IBGP. Each BGP route injected into IBGP has a specific protocol next hop. Junos OS provides the flexibility to set the protocol next hop when exporting the route into IBGP. For instance, a next-hop self can be set with an export policy configuration. You can also set the protocol next hop to a route associated with a specified topology routing table.

Keeping in mind that an EBGP route can have a community associated with a routing topology, you can conveniently configure a policy to use this community to designate which protocol next hop should be set when exporting the IBGP route into **inet.2**. As such, a specific protocol next-hop IP address is required for each topology on each router injecting IBGP routes. You can configure multiple secondary loopback IP addresses on a router to be used as protocol next-hop addresses.

A group of BGP routes associated with a routing topology use the same unique protocol next hop. For instance, if you configure a PE router to handle two routing topologies, you would also configure two unique nonprimary addresses under loopback interface lo0. Next, associate each nonprimary loopback IP address with a topology for inclusion in the associated topology routing table. Configure the loopback IP address and topology under an OSPF interface statement. You must specifically disable all other topologies known to OSPF for two reasons. First, the loopback address specific to a topology must reside in only one topology routing table. Second, once the topology is added to OSPF, the topology defaults to being enabled on all subsequent interfaces under OSPF.

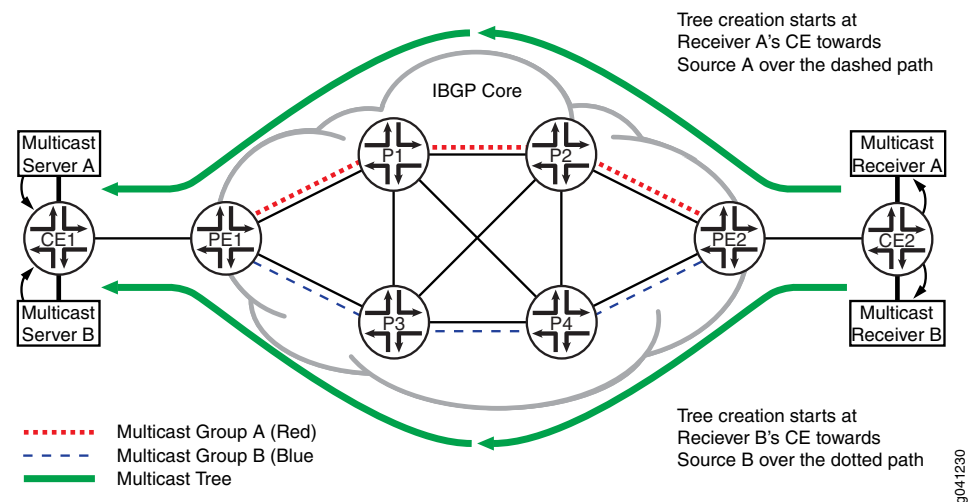
You can specify up to two routing tables in the resolution configuration. A key element to this solution is that the protocol next-hop address resides in only one topology table. That is, the protocol next hop belongs to a remote PE secondary loopback address and is injected into only one topology table. The route resolution scheme first checks the

topology table for the protocol next-hop address. If the address is found, it uses this entry. If it is not found, the resolution scheme then checks the second topology table. Hence, only one topology table is used for each protocol next-hop address.

Links can support all routing topologies to provide a backup path should a primary multicast path fail. You can configure specific OSPF link metrics on topologies to identify paths and build trees to different servers. When a multicast tree gets built with PIM join messages directed toward the source, it follows the most preferred path. A multicast tree to a different multicast source (in a different routing topology) can create another tree along a different path.

Figure 7 on page 41 shows an example of two trees using different paths over different topologies. It shows Server A using the multicast tree with the dashed line as its path and Server B using the multicast tree with the dotted line as its path.

Figure 7: Core Links Configured to Prefer Specified Routing Topologies



Example: Configuring Multitopology Routing Based on a Multicast Source

This example shows how to use multitopology routing (MTR) to provide redundancy for multicast traffic over separate network paths. That is, two multicast sources send the same multicast stream, yet for redundancy purposes in the case of link failure, the two streams use disjoint paths.



NOTE: Note there is no standard defined at this time for using MTR extensions to PIM.

- [Requirements on page 42](#)
- [Overview on page 42](#)
- [Configuration on page 42](#)
- [Verification on page 60](#)

Requirements

This example requires that Junos OS Release 9.0 or later is running on the provider core devices.

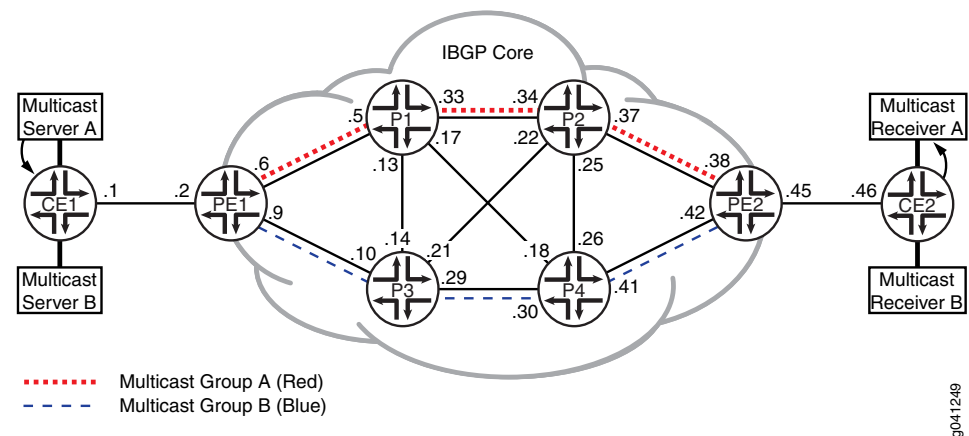
Overview

Assume that each source providing redundant multicast streams, S1 and S2, have different IP subnet addresses. Each source sends multicast traffic using different groups: G1 and G2. Further, assume that S1 and S2 are attached to the same customer edge (CE) device and use BGP to announce routes to the provider edge (PE) router.

For a complete set of configurations for all of the devices in the topology, see “[CLI Quick Configuration](#)” on page 42. The remainder of the example focuses on Device CE1 and Device PE1.

Figure 8 on page 42 shows the sample topology.

Figure 8: Multi-topology OSPF and BGP for Designating Links Belonging to Voice and Video Services



Configuration

- [Configuring Device CE1 on page 49](#)
- [Configuring Device PE1 on page 53](#)

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-0/1/0 fastether-options loopback
set interfaces fe-0/1/0 unit 0 family inet address 11.19.130.1/24
set interfaces fe-0/1/0 unit 0 family inet address 11.19.131.1/24
set interfaces fe-0/1/0 unit 0 family inet address 11.19.132.1/24
set interfaces ge-1/2/0 unit 1 description to-PE1
set interfaces ge-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces lo0 unit 97 family inet address 10.255.165.97/32 primary
set protocols bgp group ebgp type external
```



```

set protocols bgp group ebgp local-address 10.0.0.1
set protocols bgp group ebgp family inet unicast
set protocols bgp group ebgp family inet multicast
set protocols bgp group ebgp export set_community
set protocols bgp group ebgp export inject_directs
set protocols bgp group ebgp peer-as 100
set protocols bgp group ebgp neighbor 10.0.0.2
set protocols pim interface fe-0/1/0.0 mode sparse
set protocols pim interface ge-1/2/0.1 mode sparse
set policy-options policy-statement inject_directs term a from protocol direct
set policy-options policy-statement inject_directs term a from interface fe-0/1/0.0
set policy-options policy-statement inject_directs term a then next policy
set policy-options policy-statement inject_directs term a then accept
set policy-options policy-statement inject_directs term b then reject
set policy-options policy-statement set_community term a from route-filter 11.19.130.0/24
    exact
set policy-options policy-statement set_community term a from route-filter 11.19.131.0/24
    exact
set policy-options policy-statement set_community term a then community add red
set policy-options policy-statement set_community term a then accept
set policy-options policy-statement set_community term b from route-filter 11.19.132.0/24
    exact
set policy-options policy-statement set_community term b from route-filter 11.19.133.0/24
    exact
set policy-options policy-statement set_community term b then community add blue
set policy-options policy-statement set_community term b then accept
set policy-options policy-statement set_community term default then accept
set policy-options community blue members target:50:50
set policy-options community red members target:40:40
set routing-options interface-routes rib-group inet if-rib
set routing-options static route 10.0.0.0/16 next-hop 10.0.0.2
set routing-options rib-groups inet.2 import-rib inet.0
set routing-options rib-groups if-rib import-rib inet.0
set routing-options rib-groups if-rib import-rib inet.2
set routing-options rib-groups if-rib import-policy inject_directs
set routing-options autonomous-system 101

```

Device CE2

```

set interfaces fe-0/1/1 unit 0
set interfaces ge-1/2/0 unit 46 description to-PE2
set interfaces ge-1/2/0 unit 46 family inet address 10.0.0.46/30
set interfaces lo0 unit 20 family inet address 10.255.165.20/32 primary
set protocols bgp group ebgp type external
set protocols bgp group ebgp local-address 10.0.0.46
set protocols bgp group ebgp peer-as 100
set protocols bgp group ebgp neighbor 10.0.0.45
set routing-options autonomous-system 102

```

Device PE1

```

set interfaces ge-1/2/0 unit 2 description to-CE1
set interfaces ge-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces ge-1/2/1 unit 6 description to-P1
set interfaces ge-1/2/1 unit 6 family inet address 10.0.0.6/30
set interfaces ge-1/2/2 unit 9 description to-P3
set interfaces ge-1/2/2 unit 9 family inet address 10.0.0.9/30
set interfaces lo0 unit 93 family inet address 10.255.165.93/32 primary
set interfaces lo0 unit 93 family inet address 1.1.1.30/32

```

```
set interfaces lo0 unit 93 family inet address 2.2.2.30/32
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.93
set protocols bgp group ibgp family inet unicast
set protocols bgp group ibgp family inet multicast
set protocols bgp group ibgp export nhs_test
set protocols bgp group ibgp export nhs_inet0_self
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.113
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols bgp group ebgp type external
set protocols bgp group ebgp local-address 10.0.0.2
set protocols bgp group ebgp family inet unicast
set protocols bgp group ebgp family inet multicast
set protocols bgp group ebgp peer-as 101
set protocols bgp group ebgp neighbor 10.0.0.1
set protocols ospf topology red topology-id 126
set protocols ospf topology blue topology-id 52
set protocols ospf area 0.0.0.0 interface ge-1/2/1.6 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/1.6 topology blue metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/1.6 topology red
set protocols ospf area 0.0.0.0 interface ge-1/2/2.9 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/2.9 topology red metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/2.9 topology blue
set protocols ospf area 0.0.0.0 interface lo0.93 passive
set protocols ospf area 0.0.0.0 interface 1.1.1.30 topology red
set protocols ospf area 0.0.0.0 interface 1.1.1.30 topology blue disable
set protocols ospf area 0.0.0.0 interface 2.2.2.30 topology blue
set protocols ospf area 0.0.0.0 interface 2.2.2.30 topology red disable
set protocols pim rib-group inet mcast-rib
set protocols pim interface ge-1/2/0.2 mode sparse
set protocols pim interface ge-1/2/1.6 mode sparse
set protocols pim interface ge-1/2/2.9 mode sparse
set policy-options policy-statement nhs_inet0_self term a from protocol bgp
set policy-options policy-statement nhs_inet0_self term a from rib inet.0
set policy-options policy-statement nhs_inet0_self term a then next-hop self
set policy-options policy-statement nhs_test term a from protocol bgp
set policy-options policy-statement nhs_test term a from community red
set policy-options policy-statement nhs_test term a then next-hop 1.1.1.30
set policy-options policy-statement nhs_test term a then next policy
set policy-options policy-statement nhs_test term a then accept
set policy-options policy-statement nhs_test term b from protocol bgp
set policy-options policy-statement nhs_test term b from community blue
set policy-options policy-statement nhs_test term b then next-hop 2.2.2.30
set policy-options policy-statement nhs_test term b then next policy
set policy-options policy-statement nhs_test term b then accept
set policy-options policy-statement nhs_test term c then next-hop self
set policy-options community blue members target:50:50
set policy-options community red members target:40:40
set routing-options rib-groups mcast-rib import-rib inet.2
set routing-options autonomous-system 100
set routing-options resolution rib inet.2 resolution-ribs :red.inet.0
set routing-options resolution rib inet.2 resolution-ribs :blue.inet.0
set routing-options topologies family inet topology red
```

```
set routing-options topologies family inet topology blue
```

Device PE2

```
set interfaces ge-1/2/0 unit 38 description to-P2
set interfaces ge-1/2/0 unit 38 family inet address 10.0.0.38/30
set interfaces ge-1/2/1 unit 42 description to-P4
set interfaces ge-1/2/1 unit 42 family inet address 10.0.0.42/30
set interfaces ge-1/2/2 unit 45 description to-CE2
set interfaces ge-1/2/2 unit 45 family inet address 10.0.0.45/30
set interfaces lo0 unit 203 family inet address 10.255.165.203/32 primary
set interfaces lo0 unit 203 family inet address 1.1.1.40/32
set interfaces lo0 unit 203 family inet address 2.2.2.40/32
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.203
set protocols bgp group ibgp family inet unicast
set protocols bgp group ibgp family inet multicast
set protocols bgp group ibgp export nhs_test
set protocols bgp group ibgp export nhs_inet0_self
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.93
set protocols bgp group ibgp neighbor 10.255.165.113
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols bgp group ebgp type external
set protocols bgp group ebgp local-address 10.0.0.45
set protocols bgp group ebgp family inet unicast
set protocols bgp group ebgp family inet multicast
set protocols bgp group ebgp peer-as 102
set protocols bgp group ebgp neighbor 10.0.0.46
set protocols ospf topology red topology-id 126
set protocols ospf topology blue topology-id 52
set protocols ospf area 0.0.0.0 interface ge-1/2/0.38 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/0.38 topology blue metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/0.38 topology red
set protocols ospf area 0.0.0.0 interface ge-1/2/1.42 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/1.42 topology red metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/1.42 topology blue
set protocols ospf area 0.0.0.0 interface lo0.203 passive
set protocols ospf area 0.0.0.0 interface 1.1.1.40 topology red
set protocols ospf area 0.0.0.0 interface 1.1.1.40 topology blue disable
set protocols ospf area 0.0.0.0 interface 2.2.2.40 topology red disable
set protocols ospf area 0.0.0.0 interface 2.2.2.40 topology blue
set protocols pim rib-group inet mcast-rib
set protocols pim interface ge-1/2/0.38 mode sparse
set protocols pim interface ge-1/2/1.42 mode sparse
set protocols pim interface ge-1/2/2.45 mode sparse
set policy-options policy-statement nhs then next-hop self
set policy-options policy-statement nhs_inet0_self term a from protocol bgp
set policy-options policy-statement nhs_inet0_self term a from rib inet.0
set policy-options policy-statement nhs_inet0_self term a then next-hop self
set policy-options policy-statement nhs_test term a from protocol bgp
set policy-options policy-statement nhs_test term a from community red
set policy-options policy-statement nhs_test term a then next-hop 1.1.1.40
set policy-options policy-statement nhs_test term a then next policy
set policy-options policy-statement nhs_test term a then accept
set policy-options policy-statement nhs_test term b from protocol bgp
set policy-options policy-statement nhs_test term b from community blue
```

```
set policy-options policy-statement nhs_test term b then next-hop 2.2.2.40
set policy-options policy-statement nhs_test term b then next policy
set policy-options policy-statement nhs_test term b then accept
set policy-options policy-statement nhs_test term c then next-hop self
set policy-options community blue members target:50:50
set policy-options community red members target:40:40
set routing-options rib-groups mcast-rib import-rib inet.2
set routing-options autonomous-system 100
set routing-options resolution rib inet.2 resolution-ribs :red.inet.0
set routing-options resolution rib inet.2 resolution-ribs :blue.inet.0
set routing-options topologies family inet topology red
set routing-options topologies family inet topology blue
```

Device P1

```
set interfaces ge-1/2/0 unit 5 description to-PE1
set interfaces ge-1/2/0 unit 5 family inet address 10.0.0.5/30
set interfaces ge-1/2/1 unit 13 description to-P3
set interfaces ge-1/2/1 unit 13 family inet address 10.0.0.13/30
set interfaces ge-1/2/2 unit 17 description to-P4
set interfaces ge-1/2/2 unit 17 family inet address 10.0.0.17/30
set interfaces ge-1/2/3 unit 33 description to-P2
set interfaces ge-1/2/3 unit 33 family inet address 10.0.0.33/30
set interfaces lo0 unit 99 family inet address 10.255.165.99/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.99
set protocols bgp group ibgp family inet unicast
set protocols bgp group ibgp family inet multicast
set protocols bgp group ibgp neighbor 10.255.165.93
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.113
set protocols ospf topology red topology-id 126
set protocols ospf topology blue topology-id 52
set protocols ospf area 0.0.0.0 interface ge-1/2/3.33 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/3.33 topology red
set protocols ospf area 0.0.0.0 interface ge-1/2/3.33 topology blue metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/2.17
set protocols ospf area 0.0.0.0 interface ge-1/2/1.13
set protocols ospf area 0.0.0.0 interface ge-1/2/0.5 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/0.5 topology red
set protocols ospf area 0.0.0.0 interface ge-1/2/0.5 topology blue metric 1
set protocols ospf area 0.0.0.0 interface lo0.99 passive
set protocols pim rib-group inet mcast-rib
set protocols pim interface ge-1/2/3.33 mode sparse
set protocols pim interface ge-1/2/2.17 mode sparse
set protocols pim interface ge-1/2/1.13 mode sparse
set protocols pim interface ge-1/2/0.5 mode sparse
set routing-options rib-groups mcast-rib import-rib inet.2
set routing-options autonomous-system 100
set routing-options topologies family inet topology red
set routing-options topologies family inet topology blue
```

Device P2

```
set interfaces ge-1/2/0 unit 22 description to-P3
set interfaces ge-1/2/0 unit 22 family inet address 10.0.0.22/30
set interfaces ge-1/2/1 unit 25 description to-P4
```

```

set interfaces ge-1/2/1 unit 25 family inet address 10.0.0.25/30
set interfaces ge-1/2/2 unit 34 description to-P1
set interfaces ge-1/2/2 unit 34 family inet address 10.0.0.34/30
set interfaces ge-1/2/3 unit 37 description to-PE2
set interfaces ge-1/2/3 unit 37 family inet address 10.0.0.37/30
set interfaces lo0 unit 113 family inet address 10.255.165.113/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.113
set protocols bgp group ibgp family inet unicast
set protocols bgp group ibgp family inet multicast
set protocols bgp group ibgp neighbor 10.255.165.93
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols ospf topology red topology-id 126
set protocols ospf topology blue topology-id 52
set protocols ospf area 0.0.0.0 interface ge-1/2/2.34 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/2.34 topology red
set protocols ospf area 0.0.0.0 interface ge-1/2/2.34 topology blue metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/0.22
set protocols ospf area 0.0.0.0 interface ge-1/2/1.25
set protocols ospf area 0.0.0.0 interface ge-1/2/3.37 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/3.37 topology red
set protocols ospf area 0.0.0.0 interface ge-1/2/3.37 topology blue metric 1
set protocols ospf area 0.0.0.0 interface lo0.113 passive
set protocols pim rib-group inet mcast-rib
set protocols pim interface ge-1/2/2.34 mode sparse
set protocols pim interface ge-1/2/0.22 mode sparse
set protocols pim interface ge-1/2/1.25 mode sparse
set protocols pim interface ge-1/2/3.37 mode sparse
set routing-options rib-groups mcast-rib import-rib inet.2
set routing-options autonomous-system 100
set routing-options topologies family inet topology red
set routing-options topologies family inet topology blue

```

Device P3

```

set interfaces ge-1/2/0 unit 10 description to-PE1
set interfaces ge-1/2/0 unit 10 family inet address 10.0.0.10/30
set interfaces ge-1/2/1 unit 14 description to-P1
set interfaces ge-1/2/1 unit 14 family inet address 10.0.0.14/30
set interfaces ge-1/2/2 unit 21 description to-P2
set interfaces ge-1/2/2 unit 21 family inet address 10.0.0.21/30
set interfaces ge-1/2/3 unit 29 description to-P4
set interfaces ge-1/2/3 unit 29 family inet address 10.0.0.29/30
set interfaces lo0 unit 111 family inet address 10.255.165.111/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.111
set protocols bgp group ibgp family inet unicast
set protocols bgp group ibgp family inet multicast
set protocols bgp group ibgp neighbor 10.255.165.93
set protocols bgp group ibgp neighbor 10.255.165.113
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.95
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols ospf topology red topology-id 126
set protocols ospf topology blue topology-id 52

```

```
set protocols ospf area 0.0.0.0 interface ge-1/2/3.29 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/3.29 topology red metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/3.29 topology blue
set protocols ospf area 0.0.0.0 interface ge-1/2/2.21
set protocols ospf area 0.0.0.0 interface ge-1/2/1.14
set protocols ospf area 0.0.0.0 interface ge-1/2/0.10 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/0.10 topology red metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/0.10 topology blue
set protocols ospf area 0.0.0.0 interface lo0.111 passive
set protocols pim rib-group inet mcast-rib
set protocols pim interface ge-1/2/3.29 mode sparse
set protocols pim interface ge-1/2/2.21 mode sparse
set protocols pim interface ge-1/2/1.14 mode sparse
set protocols pim interface ge-1/2/0.10 mode sparse
set routing-options rib-groups mcast-rib import-rib inet.2
set routing-options autonomous-system 100
set routing-options topologies family inet topology red
set routing-options topologies family inet topology blue
```

Device P4

```
set interfaces ge-1/2/0 unit 18 description to-P1
set interfaces ge-1/2/0 unit 18 family inet address 10.0.0.18/30
set interfaces ge-1/2/1 unit 26 description to-P2
set interfaces ge-1/2/1 unit 26 family inet address 10.0.0.26/30
set interfaces ge-1/2/2 unit 30 description to-P3
set interfaces ge-1/2/2 unit 30 family inet address 10.0.0.30/30
set interfaces ge-1/2/3 unit 41 description to-PE2
set interfaces ge-1/2/3 unit 41 family inet address 10.0.0.41/30
set interfaces lo0 unit 95 family inet address 10.255.165.95/32 primary
set protocols bgp group ibgp type internal
set protocols bgp group ibgp local-address 10.255.165.95
set protocols bgp group ibgp family inet unicast
set protocols bgp group ibgp family inet multicast
set protocols bgp group ibgp neighbor 10.255.165.93
set protocols bgp group ibgp neighbor 10.255.165.113
set protocols bgp group ibgp neighbor 10.255.165.203
set protocols bgp group ibgp neighbor 10.255.165.111
set protocols bgp group ibgp neighbor 10.255.165.99
set protocols ospf topology red topology-id 126
set protocols ospf topology blue topology-id 52
set protocols ospf area 0.0.0.0 interface ge-1/2/2.30 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/2.30 topology red metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/2.30 topology blue
set protocols ospf area 0.0.0.0 interface ge-1/2/0.18
set protocols ospf area 0.0.0.0 interface ge-1/2/1.26
set protocols ospf area 0.0.0.0 interface ge-1/2/3.41 metric 10
set protocols ospf area 0.0.0.0 interface ge-1/2/3.41 topology red metric 1
set protocols ospf area 0.0.0.0 interface ge-1/2/3.41 topology blue
set protocols ospf area 0.0.0.0 interface lo0.95 passive
set protocols pim rib-group inet mcast-rib
set protocols pim interface ge-1/2/2.30 mode sparse
set protocols pim interface ge-1/2/0.18 mode sparse
set protocols pim interface ge-1/2/1.26 mode sparse
set protocols pim interface ge-1/2/3.41 mode sparse
set routing-options rib-groups mcast-rib import-rib inet.2
set routing-options autonomous-system 100
set routing-options topologies family inet topology red
```

```
set routing-options topologies family inet topology blue
```

Configuring Device CE1

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 CE1:

1. Configure the interfaces.

For demonstration purposes, the example places an Ethernet interface into loopback mode and configures several addresses on this loopback interface. The addresses are then announced to the network as direct routes. These routes simulate a group of BGP routes with communities attached.

```
[edit interfaces]
user@CE1# set fe-0/1/0 fastether-options loopback
user@CE1# set fe-0/1/0 unit 0 family inet address 11.19.130.1/24
user@CE1# set fe-0/1/0 unit 0 family inet address 11.19.131.1/24
user@CE1# set fe-0/1/0 unit 0 family inet address 11.19.132.1/24
```

```
user@CE1# set ge-1/2/0 unit 1 description to-PE1
user@CE1# set ge-1/2/0 unit 1 family inet address 10.0.0.1/30
```

```
user@CE1# set lo0 unit 97 family inet address 10.255.165.97/32 primary
```

2. Configure the external BGP (EBGP) connection to Device PE1.

The CE router nearest to the multicast servers announces the multicast source IP addresses to the PE routers using EBGP. The source addresses are announced with both **family inet unicast** and **family inet multicast**, thus causing the BGP route to be added to the default routing table, **inet.0**, and to the multicast routing table, **inet.2**. Both sets of routes are injected by the PE router into IBGP.

```
[edit protocols bgp group ebgp]
user@CE1# set type external
user@CE1# set local-address 10.0.0.1
user@CE1# set family inet unicast
user@CE1# set family inet multicast
user@CE1# set peer-as 100
user@CE1# set neighbor 10.0.0.2
```

3. Configure PIM on the interfaces.

```
[edit protocols pim]
user@CE1# set interface fe-0/1/0.0 mode sparse
user@CE1# set interface ge-1/2/0.1 mode sparse
```

4. Configure the routing policy that announces the addresses that are configured on interface fe-0/1/0.

```
[edit policy-options policy-statement inject_directs]
user@CE1# set term a from protocol direct
user@CE1# set term a from interface fe-0/1/0.0
user@CE1# set term a then next policy
```

```
user@CE1# set term a then accept
user@CE1# set term b then reject
```

5. Configure the routing policy that tags some routes with the red community attribute and other routes with the blue community attribute.

The CE router advertises routes through EBGp to the PE router. These routes are advertised as BGP **family inet multicast** routes with communities set for two different groups. Policies identify the two groups of BGP routes.

```
[edit policy-options policy-statement set_community term a]
user@CE1# set from route-filter 11.19.130.0/24 exact
user@CE1# set from route-filter 11.19.131.0/24 exact
user@CE1# set then community add red
user@CE1# set then accept
```

```
[edit policy-options policy-statement set_community term b]
user@CE1# set from route-filter 11.19.132.0/24 exact
user@CE1# set from route-filter 11.19.133.0/24 exact
user@CE1# set then community add blue
user@CE1# set then accept
```

```
[edit policy-options policy-statement set_community term default]
user@CE1# set then accept
```

```
[edit policy-options]
user@CE1# set community blue members target:50:50
user@CE1# set community red members target:40:40
```

6. Apply the **set_community** export policy so that the direct routes are exported into BGP.

Apply the **inject_directs** export policy to announce the addresses that are configured on interface fe-0/1/0.

```
[edit protocols bgp group ebgp]
user@CE1# set export set_community
user@CE1# set export inject_directs
```

7. Use **rib-groups** to simulate a group of BGP routes with communities attached and announced as multicast routes.

This configuration creates a multicast routing table and causes PIM to use the multicast routing table **inet.2**.

```
[edit routing-options]
user@CE1# set interface-routes rib-group inet if-rib
```

```
user@CE1# set static route 10.0.0.0/16 next-hop 10.0.0.2
```

```
[edit routing-options rib-groups]
user@CE1# set inet.2 import-rib inet.0
user@CE1# set if-rib import-rib inet.0
user@CE1# set if-rib import-rib inet.2
user@CE1# set if-rib import-policy inject_directs
```


8. Configure the autonomous system (AS) number.

```
[edit routing-options]
user@CE1# set autonomous-system 101
```

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@CE1# show interfaces
fe-0/1/0 {
  fastether-options {
    loopback;
  }
  unit 0 {
    family inet {
      address 11.19.130.1/24;
      address 11.19.131.1/24;
      address 11.19.132.1/24;
    }
  }
}
ge-1/2/0 {
  unit 1 {
    description to-PE1;
    family inet {
      address 10.0.0.1/30;
    }
  }
}
lo0 {
  unit 97 {
    family inet {
      address 10.255.165.97/32 {
        primary;
      }
    }
  }
}

user@CE1# show protocols
bgp {
  group ebgp {
    type external;
    local-address 10.0.0.1;
    family inet {
      unicast;
      multicast;
    }
    export [ set_community inject_directs ];
    peer-as 100;
    neighbor 10.0.0.2;
  }
}
pim {
```

```
interface fe-0/1/0.0 {
  mode sparse;
}
interface ge-1/2/0.1 {
  mode sparse;
}
}

user@CE1# show policy-options
policy-statement inject_directs {
  term a {
    from {
      protocol direct;
      interface fe-0/1/0.0;
    }
    then {
      next policy;
      accept;
    }
  }
  term b {
    then reject;
  }
}
policy-statement set_community {
  term a {
    from {
      route-filter 11.19.130.0/24 exact;
      route-filter 11.19.131.0/24 exact;
    }
    then {
      community add red;
      accept;
    }
  }
  term b {
    from {
      route-filter 11.19.132.0/24 exact;
      route-filter 11.19.133.0/24 exact;
    }
    then {
      community add blue;
      accept;
    }
  }
  term default {
    then accept;
  }
}
community blue members target:50:50;
community red members target:40:40;

user@CE1# show routing-options
interface-routes {
  rib-group inet if-rib;
}
static {
```

```

    route 10.0.0.0/16 next-hop 10.0.0.2;
  }
  rib-groups {
    inet.2 {
      import-rib inet.0;
    }
    if-rib {
      import-rib [ inet.0 inet.2 ];
      import-policy inject_directs;
    }
  }
  autonomous-system 101;

```

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

Configuring Device PE1

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 PE1:

1. Configure the interfaces.

```

[edit interfaces]
user@PE1# set ge-1/2/0 unit 2 description to-CE1
user@PE1# set ge-1/2/0 unit 2 family inet address 10.0.0.2/30

user@PE1# set ge-1/2/1 unit 6 description to-P1
user@PE1# set ge-1/2/1 unit 6 family inet address 10.0.0.6/30

user@PE1# set ge-1/2/2 unit 9 description to-P3
user@PE1# set ge-1/2/2 unit 9 family inet address 10.0.0.9/30

user@PE1# set lo0 unit 93 family inet address 10.255.165.93/32 primary

```

2. Configure secondary addresses, 1.1.1.30 and 2.2.2.30.

A specific protocol next-hop IP address is required for each topology on each router injecting IBGP routes. You can configure multiple secondary loopback IP addresses on a router to be used as protocol next-hop addresses. This configuration shows nonprimary IP addresses 1.1.1.30/32 and 2.2.2.30/32 configured on loopback interface lo0 for use in the red and blue topologies, respectively.

A group of BGP routes associated with a routing topology use the same unique protocol next hop. For instance, if you configure a PE router to handle two routing topologies, then you would also configure two unique nonprimary addresses under loopback interface lo0.

```

[edit interfaces]
user@PE1# set lo0 unit 93 family inet address 1.1.1.30/32
user@PE1# set lo0 unit 93 family inet address 2.2.2.30/32

```

3. Associate each nonprimary loopback IP address with a topology for inclusion in the associated topology routing table.

Configure the loopback IP address and topology under an OSPF interface statement. You must specifically disable all other topologies known to OSPF for two reasons. First, the loopback address specific to a topology must reside in only one topology routing table. Second, once the topology is added to OSPF, the topology defaults to being enabled on all subsequent interfaces under OSPF.

The Device PE1 configuration places the loopback address 1.1.1.30/32 into the OSPF database as a stub route under this router's OSPF Router-LSA. It belongs to the red and default topologies, but not to the blue topology. The loopback address 1.1.1.30/32 is installed in the remote core routers' topology routing tables **inet.0** and **:red.inet.0**, (but not in **:blue.inet.0**). Use a similar configuration for the blue loopback address 2.2.2.30/32.

```
[edit protocols ospf]
user@PE1# set topology red topology-id 126
user@PE1# set topology blue topology-id 52

[edit protocols ospf area 0.0.0.0]
user@PE1# set interface 1.1.1.30 topology red
user@PE1# set interface 1.1.1.30 topology blue disable

user@PE1# set interface 2.2.2.30 topology blue
user@PE1# set interface 2.2.2.30 topology red disable
```

4. Enable OSPF on the interfaces, and configure specific OSPF link metrics on topologies to identify paths and build trees to different servers.

Links can support all routing topologies to provide backup should a primary multicast path fail.

When a multicast tree gets built through PIM join messages directed toward the source, it follows the most preferred path. A multicast tree to a different multicast source (in a different routing topology) can create another tree along a different path.

```
[edit protocols ospf area 0.0.0.0]
user@PE1# set interface ge-1/2/1.6 metric 10
user@PE1# set interface ge-1/2/1.6 topology blue metric 1
user@PE1# set interface ge-1/2/1.6 topology red

user@PE1# set interface ge-1/2/2.9 metric 10
user@PE1# set interface ge-1/2/2.9 topology red metric 1
user@PE1# set interface ge-1/2/2.9 topology blue

user@PE1# set interface lo0.93 passive
```

5. Create the multicast routing table **inet.2**, and configure PIM to use the **inet.2** routing table.

Set up a separate routing table for multicast lookups. It is populated with routes from **inet.2**. The **inet.2** routing table is populated by routes of type multicast.

```
[edit routing-options]
user@PE1# set rib-groups mcast-rib import-rib inet.2
```

6. Configure PIM to use the routes in **inet.2**.

```
[edit protocols pim]
user@PE1# set rib-group inet mcast-rib
```

7. Enable PIM on the interfaces.

```
[edit protocols pim]
user@PE1# set interface ge-1/2/0.2 mode sparse
user@PE1# set interface ge-1/2/1.6 mode sparse
user@PE1# set interface ge-1/2/2.9 mode sparse
```

8. Configure the router to perform route resolution on protocol next hops using specified routing tables.

The protocol next hop is used to determine the forwarding next-hop interface out of which to forward PIM join messages. This configuration directs **inet.2** route resolution to use topology routing tables **:red.inet.0** and **:blue.inet.0** for protocol next-hop IP address lookups.

You can specify up to two routing tables in the resolution configuration. A key element to this solution is that the protocol next-hop address resides in only one topology routing table. That is, the protocol next hop belongs to a remote PE secondary loopback address and is injected into only one topology routing table. The route resolution scheme first checks routing table **:red.inet.0** for the protocol next-hop address. If the address is found, it uses this entry. If it is not found, the resolution scheme checks routing table **:blue.inet.0**. Hence, only one topology routing table is used for each protocol nexthop address.

```
[edit routing-options resolution rib inet.2]
user@PE1# set resolution-ribs :red.inet.0
user@PE1# set resolution-ribs :blue.inet.0
```

9. Configure the autonomous system (AS) number.

```
[edit routing-options]
user@PE1# set autonomous-system 100
```

10. Configure BGP.

```
[edit protocols bgp group ibgp]
user@PE1# set type internal
user@PE1# set local-address 10.255.165.93
user@PE1# set family inet unicast
user@PE1# set family inet multicast
user@PE1# set neighbor 10.255.165.111
user@PE1# set neighbor 10.255.165.203
user@PE1# set neighbor 10.255.165.113
user@PE1# set neighbor 10.255.165.95
user@PE1# set neighbor 10.255.165.99
```

```
[edit protocols bgp group ebgp]
user@PE1# set type external
user@PE1# set local-address 10.0.0.2
user@PE1# set family inet unicast
user@PE1# set family inet multicast
user@PE1# set peer-as 101
user@PE1# set neighbor 10.0.0.1
```

11. Set the protocol next hop when exporting EBGp routes into IBGP.

Configure the ingress Device PE1 router to set the BGP route's protocol next-hop address when exporting the route into IBGP.

BGP uses an export policy to set the next hop when injecting the EBGp routes into IBGP.

This configuration is an export policy where there are three possibilities of next hops being set. Route 1.1.1.30 is associated with the red topology. Route 2.2.2.30 is associated with the blue topology. For the default next-hop self policy, the primary loopback address 10.255.165.93 on Device PE1 is used.

The **nhs_test** policy sets the protocol next-hop based on the community in the BGP update.

```
[edit policy-options]
user@PE1# set community blue members target:50:50
user@PE1# set community red members target:40:40
```

```
[edit policy-options policy-statement nhs_test term a]
user@PE1# set from protocol bgp
user@PE1# set from community red
user@PE1# set then next-hop 1.1.1.30
user@PE1# set then next policy
user@PE1# set then accept
```

```
[edit policy-options policy-statement nhs_test term b]
user@PE1# set from protocol bgp
user@PE1# set from community blue
user@PE1# set then next-hop 2.2.2.30
user@PE1# set then next policy
user@PE1# set then accept
user@PE1# set policy-options policy-statement nhs_test term c then next-hop self
```

```
[edit policy-options policy-statement nhs_inet0_self term a]
user@PE1# set from protocol bgp
user@PE1# set from rib inet.0
user@PE1# set then next-hop self
```

12. Apply the next-hop self policies to the IBGP sessions.

```
[edit protocols bgp group ibgp]
user@PE1# set export nhs_test
user@PE1# set export nhs_inet0_self
```

13. Configure the voice and video topologies, which enable you to use these topologies with OSPF and BGP.

The names **voice** and **video** are local to the router. The names are not propagated beyond this router. However, for management purposes, a consistent naming scheme across routers in a multitopology environment is convenient.

```
[edit routing-options topologies family inet]
user@PE1# set topology red
user@PE1# set topology blue
```

Results From configuration mode, confirm your configuration by entering the **show interfaces**, **show protocols**, and **show routing-options**, 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@PE1# show interfaces
ge-1/2/0 {
  unit 2 {
    description to-CE1;
    family inet {
      address 10.0.0.2/30;
    }
  }
}
ge-1/2/1 {
  unit 6 {
    description to-P1;
    family inet {
      address 10.0.0.6/30;
    }
  }
}
ge-1/2/2 {
  unit 9 {
    description to-P3;
    family inet {
      address 10.0.0.9/30;
    }
  }
}
lo0 {
  unit 93 {
    family inet {
      address 10.255.165.93/32 {
        primary;
      }
      address 1.1.1.30/32;
      address 2.2.2.30/32;
    }
  }
}

user@PE1# show protocols
bgp {
  group ibgp {
    type internal;
    local-address 10.255.165.93;
    family inet {
      unicast;
      multicast;
    }
    export [ nhs_test nhs_inet0_self ];
    neighbor 10.255.165.111;
    neighbor 10.255.165.203;
    neighbor 10.255.165.113;
    neighbor 10.255.165.95;
  }
}

```

```
        neighbor 10.255.165.99;
    }
    group ebgp {
        type external;
        local-address 10.0.0.2;
        family inet {
            unicast;
            multicast;
        }
        peer-as 101;
        neighbor 10.0.0.1;
    }
}
ospf {
    topology red topology-id 126;
    topology blue topology-id 52;
    area 0.0.0.0 {
        interface ge-1/2/1.6 {
            metric 10;
            topology blue metric 1;
            topology red;
        }
        interface ge-1/2/2.9 {
            metric 10;
            topology red metric 1;
            topology blue;
        }
        interface lo0.93 {
            passive;
        }
        interface 1.1.1.30 {
            topology red;
            topology blue disable;
        }
        interface 2.2.2.30 {
            topology blue;
            topology red disable;
        }
    }
}
pim {
    rib-group inet mcast-rib;
    interface ge-1/2/0.2 {
        mode sparse;
    }
    interface ge-1/2/1.6 {
        mode sparse;
    }
    interface ge-1/2/2.9 {
        mode sparse;
    }
}
user@PE1# show policy-options
policy-statement nhs_inet0_self {
    term a {
```



```

        from {
            protocol bgp;
            rib inet.0;
        }
        then {
            next-hop self;
        }
    }
}
policy-statement nhs_test {
    term a {
        from {
            protocol bgp;
            community red;
        }
        then {
            next-hop 1.1.1.30;
            next policy;
            accept;
        }
    }
    term b {
        from {
            protocol bgp;
            community blue;
        }
        then {
            next-hop 2.2.2.30;
            next policy;
            accept;
        }
    }
    term c {
        then {
            next-hop self;
        }
    }
}
community blue members target:50:50;
community red members target:40:40;

user@PE1# show routing-options
rib-groups {
    mcast-rib {
        import-rib inet.2;
    }
}
autonomous-system 100;
resolution {
    rib inet.2 {
        resolution-ribs [ :red.inet.0 :blue.inet.0 ];
    }
}
topologies {
    family inet {
        topology red;
    }
}

```

```
        topology blue;
    }
}
```

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

Verification

Confirm that the configuration is working properly.

- [Checking the IBGP routes in inet.2 on page 60](#)
- [Verifying the Routes on page 61](#)
- [Checking the Resolving BGP Next Hops on page 62](#)
- [Examining the Protocol Next Hop on page 63](#)
- [Verifying the OSPF Neighbor on page 64](#)
- [Checking the Router LSA on page 64](#)
- [Checking How Traffic Traverses the Network on page 65](#)

Checking the IBGP routes in inet.2

Purpose Make sure that the routes injected into IBGP by Device PE1 have next hops that are based on the topology to which they belong.

Action From operational mode, enter the **show route table extensive** command.

```
user@PE1> show route 11.19.130.0/24 table inet.2 extensive
inet.2: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
11.19.130.0/24 (1 entry, 1 announced)
TSI:
Page 0 idx 0 Type 1 val 93e9768
  Flags: Nexthop Change
  Nexthop: 1.1.1.30
  Localpref: 100
  AS path: [100] 101 I
  Communities: target:40:40
Path 11.19.130.0 from 10.0.0.1 Vector len 4. Val: 0
  *BGP Preference: 170/-101
    Next hop type: Router, Next hop index: 1180
    Address: 0x94003ec
    Next-hop reference count: 16
    Source: 10.0.0.1
    Next hop: 10.0.0.1 via lt-1/2/0.2, selected
    Session Id: 0x380004
    State: <Active Ext>
    Local AS: 100 Peer AS: 101
    Age: 22
    Validation State: unverified
    Task: BGP_101.10.0.0.1+58346
    Announcement bits (1): 0-BGP_RT_Background
    AS path: 101 I
    Communities: target:40:40
    Accepted
    Localpref: 100
    Router ID: 10.255.165.97
```

Meaning This output shows an IBGP route in the **inet.2** routing table, as seen from Device PE1. The route was originally injected into IBGP by Device PE1, where the next hop was set based on the topology to which the route belonged. The BGP community value determined the topology association.

The route 11.19.130/24 belongs to the red topology because it has a community value of target:40:40. The protocol next hop is 1.1.1.30, and the forwarding next hop is ge-1/2/1.42.

Verifying the Routes

Purpose Make sure that the routes are in the expected routing tables and that the expected communities are attached to the routes.

Action From operational mode, enter the **show route detail** command on Device PE1.

```
user@PE1> show route 11.19.130.0/24 detail
```

```
inet.0: 29 destinations, 30 routes (29 active, 0 holddown, 0 hidden)
11.19.130.0/24 (1 entry, 1 announced)
  *BGP      Preference: 170/-101
            Next hop type: Router, Next hop index: 812
            Address: 0xb9f064c
            Next-hop reference count: 22
            Source: 10.0.0.1
            Next hop: 10.0.0.1 via fe-1/2/0.2, selected
            Session Id: 0x600004
            State: <Active Ext>
            Local AS: 100 Peer AS: 101
            Age: 3d 21:44:07
            Task: BGP_101.10.0.0.1+51873
            Announcement bits (3): 0-KRT 3-BGP_RT_Background 4-Resolve tree
3
            AS path: 101 I
            Communities: target:40:40
            Accepted
            Localpref: 100
            Router ID: 10.255.165.97
            Secondary Tables: :voice.inet.0

:voice.inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
11.19.130.0/24 (1 entry, 1 announced)
  *BGP      Preference: 170/-101
            Next hop type: Router, Next hop index: 812
            Address: 0xb9f064c
            Next-hop reference count: 22
            Source: 10.0.0.1
            Next hop: 10.0.0.1 via fe-1/2/0.2, selected
            Session Id: 0x600004
            State: <Secondary Active IndepResolution Ext>
            Local AS: 100 Peer AS: 101
            Age: 3d 21:44:07
            Task: BGP_101.10.0.0.1+51873
            Announcement bits (2): 0-KRT 1-Resolve tree 1
            AS path: 101 I
            Communities: target:40:40
            Accepted
            Localpref: 100
```

```
Router ID: 10.255.165.97
Primary Routing Table inet.0
```

Meaning This output shows BGP route 11.19.130.0/24 with community value target:40:40. Because the route matches the criteria for the voice topology, it is added to both the default and voice topology routing tables (**inet.0** and **:voice.inet.0**). Device PE1 learns the route from Device CE1 through EBGp and then injects the route into IBGP.

Checking the Resolving BGP Next Hops

Purpose Check the protocol next hop and forwarding next hop.

Action From operational mode, enter the **show route detail** command on Device PE2.

```
user@PE2> show route 11.19.130.0/24 detail
inet.0: 29 destinations, 30 routes (29 active, 0 holddown, 0 hidden)
11.19.130.0/24 (1 entry, 1 announced)
  *BGP      Preference: 170/-101
            Next hop type: Indirect
            Address: 0xb9f0e04
            Next-hop reference count: 12
            Source: 10.255.165.93
            Next hop type: Router, Next hop index: 262153
            Next hop: 10.0.0.37 via fe-1/2/0.38
            Session Id: 0x700004
            Next hop: 10.0.0.41 via fe-1/2/1.42, selected
            Session Id: 0x700005
            Protocol next hop: 10.255.165.93
            Indirect next hop: bb8c000 262154 INH Session ID: 0x700007
            State: <Active Int Ext>
            Local AS: 100 Peer AS: 100
            Age: 3d 4:27:40 Metric2: 30
            Task: BGP_100.10.255.165.93+179
            Announcement bits (3): 0-KRT 3-BGP_RT_Background 4-Resolve tree
3
      AS path: 101 I
      Communities: target:40:40
      Accepted
      Localpref: 100
      Router ID: 10.255.165.93
      Secondary Tables: :voice.inet.0

:voice.inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
11.19.130.0/24 (1 entry, 1 announced)
  *BGP      Preference: 170/-101
            Next hop type: Indirect
            Address: 0xb9f0f34
            Next-hop reference count: 6
            Source: 10.255.165.93
            Next hop type: Router, Next hop index: 1188
            Next hop: 10.0.0.37 via fe-1/2/0.38, selected
            Session Id: 0x700004
            Protocol next hop: 10.255.165.93
            Indirect next hop: bb8c1d8 262177 INH Session ID: 0x700007
            State: <Secondary Active IndepResolution Int Ext>
            Local AS: 100 Peer AS: 100
            Age: 3d 2:00:20 Metric2: 30
            Task: BGP_100.10.255.165.93+179
```

```

Announcement bits (2): 0-KRT 1-Resolve tree 1
AS path: 101 I
Communities: target:40:40
Accepted
Localpref: 100
Router ID: 10.255.165.93
Primary Routing Table inet.0

```

Meaning A typical IBGP core has BGP routes with protocol next hops that resolve using the underlying IGP routes. IBGP routes in a topology routing table have protocol next-hop IP addresses. By default, the same topology routing table is used to look up and resolve the protocol next-hop IP address to a forwarding next hop. This output from Device PE2 shows the same BGP route as seen in the previous example: 11.19.130.0/24. The route is being shown from a different perspective, that is, from Device PE2 as an IBGP route. Similarly, this IBGP route is added to both **inet.0** and **:voice.inet.0** on Device PE2. However, while each route has the same protocol next hop, each route has a different forwarding next hop (ge-0/0/3.0 instead of ge-0/1/4.0). The reason for this difference is when the protocol next-hop IP address 10.255.165.93 is resolved, it uses the corresponding routing table (**inet.0** or **:voice.inet.0**) to look up the protocol next hop.

Examining the Protocol Next Hop

Purpose Check the protocol next hop and forwarding next hop.

Action From operational mode, enter the **show route** command on Device PE2.

```

user@PE2> show route 10.255.165.93
inet.0: 29 destinations, 30 routes (29 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.165.93/32    *[OSPF/10] 3d 04:37:26, metric 30
                   > to 10.0.0.37 via fe-1/2/0.38
                   to 10.0.0.41 via fe-1/2/1.42

:voice.inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.165.93/32    *[OSPF/10] 3d 02:10:04, metric 30
                   > to 10.0.0.37 via fe-1/2/0.38

:video.inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.165.93/32    *[OSPF/10] 3d 02:03:16, metric 30
                   > to 10.0.0.41 via fe-1/2/1.42

```

Meaning This output from Device PE2 shows the protocol next hop of 11.19.130.0/24, which is IP address 10.255.165.93, thus further demonstrating how IBGP route 11.19.130.0/24 resolves its protocol next hop. The forwarding next hops of 10.255.165.93 match the IBGP forwarding next hops of route 11.19.130/24 as shown in the previous example. Observe here that the IP address 10.255.165.93 is also in routing table **:video.inet.0**. This address is the loopback address of Device PE1, and as such, resides in all three routing tables. This example also shows how traffic entering Device PE2 destined to 11.19.130.0/24 exits different interfaces depending on its associated topology. The actual traffic is marked

in such a way that a firewall filter can direct the traffic to use a particular topology routing table.

Verifying the OSPF Neighbor

Purpose Make sure that the expected topologies are enabled on the OSPF neighbor.

Action From operational mode, enter the `show (ospf | ospf3) neighbor extensive` command on Device P2.

```
user@P2> show ospf neighbor 10.0.0.21 extensive
Address          Interface          State    ID                Pri  Dead
10.0.0.21        fe-1/2/0.22       Full    10.255.165.111   128   39
Area 0.0.0.0, opt 0x52, DR 10.0.0.22, BDR 10.0.0.21
Up 3d 06:09:50, adjacent 3d 06:09:50
Topology default (ID 0) -> Bidirectional
Topology video (ID 52) -> Bidirectional
```

Meaning This Device P2 output shows OSPF neighbor PE2 (10.0.0.21), where multitopology OSPF default and video are participants. The **Bidirectional** flag shows that the neighbor is configured using the same multitopology OSPF ID.

Checking the Router LSA

Purpose Check the links where video and voice topologies are enabled.

Action From operational mode, enter the `show ospf database extensive` command on Device P2.

```
user@P2> show ospf database lsa-id 10.255.165.203 extensive

OSPF database, Area 0.0.0.0
Type      ID                Adv Rtr          Seq      Age  Opt  Cksum  Len
Router    10.255.165.203    10.255.165.203   0x80000063   1552  0x22 0xdff3   80
bits 0x0, link count 3
id 10.255.165.203, data 255.255.255.255, Type Stub (3)
Topology count: 2, Default metric: 0
Topology video (ID 52) -> Metric: 0
Topology voice (ID 126) -> Metric: 0
id 10.0.0.38, data 10.0.0.38, Type Transit (2)
Topology count: 2, Default metric: 10
Topology video (ID 52) -> Metric: 200
Topology voice (ID 126) -> Metric: 10
id 10.0.0.42, data 10.0.0.42, Type Transit (2)
Topology count: 1, Default metric: 10
Topology video (ID 52) -> Metric: 10
Topology default (ID 0)
Type: Transit, Node ID: 10.0.0.42
Metric: 10, Bidirectional
Type: Transit, Node ID: 10.0.0.38
Metric: 10, Bidirectional
Topology video (ID 52)
Type: Transit, Node ID: 10.0.0.42
Metric: 10, Bidirectional
Type: Transit, Node ID: 10.0.0.38
Metric: 200, Bidirectional
Topology voice (ID 126)
```

```

Type: Transit, Node ID: 10.0.0.38
Metric: 10, Bidirectional
Aging timer 00:34:08
Installed 00:25:49 ago, expires in 00:34:08, sent 00:25:47 ago
Last changed 3d 01:45:51 ago, Change count: 10

```

Meaning This Device P2 output shows the Router-LSA originated by Device PE2. The LSA shows links where video and voice topologies are enabled (in addition to the default topology).

Checking How Traffic Traverses the Network

Purpose Make sure that the expected paths are used.

Action From operational mode, enter the `traceroute` command on Device CE1.

The first example output shows that a traceroute over the voice topology goes from Device CE1 to Device CE2 where DSCPs are set. The routes are resolved over `:voice.inet.0`. This traceroute path follows the voice path CE1-PE1-P1-P2-PE2-CE2.

```

user@CE1> traceroute 11.19.140.1 source 11.19.130.1 tos 160
traceroute to 11.19.140.1 (11.19.140.1) from 11.19.130.1, 30 hops max, 40 byte
packets
 1 10.0.0.2 (10.0.0.2) 2.015 ms 1.924 ms 1.770 ms
 2 10.0.0.5 (10.0.0.5) 1.890 ms 1.010 ms 0.974 ms
 3 10.0.0.34 (10.0.0.34) 0.986 ms 1.031 ms 0.973 ms
 4 10.0.0.38 (10.0.0.38) 1.213 ms 1.065 ms 1.154 ms
 5 11.19.140.1 (11.19.140.1) 1.696 ms 4.286 ms 1.332 ms

```

This output shows a traceroute from Device CE1 to Device CE2 for voice where no DSCPs are set. The routes are resolved over `inet.0`, and the resulting path is different from the previous case where the DSCPs are set. This traceroute path follows the default path CE1-PE1-P4-PE2-CE2.

```

user@CE1> traceroute 11.19.140.1 source 11.19.130.1

traceroute to 11.19.140.1 (11.19.140.1) from 11.19.130.1, 30 hops max, 40 byte
packets
 1 10.0.0.2 (10.0.0.2) 1.654 ms 1.710 ms 1.703 ms
 2 10.0.0.5 (10.0.0.5) 1.790 ms 1.045 ms 0.975 ms
 3 10.0.0.18 (10.0.0.18) 0.989 ms 1.041 ms 0.983 ms
 4 10.0.0.42 (10.0.0.42) 0.994 ms 1.036 ms 1.002 ms
 5 11.19.140.1 (11.19.140.1) 1.329 ms 2.248 ms 2.225 ms

```

This output shows a traceroute from Device CE1 to Device CE2 for video traffic where the firewall filter is based on the destination address. The routes are resolved over `:video.inet.0`. This traceroute follows the video path CE1-PE1-P3-P4-PE2-CE2.

```

user@CE1> traceroute 11.19.142.1 source 11.19.132.1

traceroute to 11.19.142.1 (11.19.142.1) from 11.19.132.1, 30 hops max, 40 byte
packets
 1 10.0.0.2 (10.0.0.2) 1.126 ms 1.300 ms 0.995 ms
 2 10.0.0.10 (10.0.0.10) 0.981 ms 1.018 ms 0.991 ms
 3 10.0.0.30 (10.0.0.30) 0.997 ms 1.886 ms 1.952 ms

```

```
4 10.0.0.42 (10.0.0.42) 1.800 ms 1.038 ms 0.980 ms
5 11.19.142.1 (11.19.142.1) 1.367 ms 1.352 ms 1.328 ms
```

This output shows a traceroute from Device CE1 to Device CE2 for video where DSCPs are set. The DSCP bits are directing Device PE1 to use the topology table **:voice.inet.0**. Because there is no entry in the voice routing table for the video routes, traffic is dropped.

```
user@CE1> traceroute 11.19.142.1 source 11.19.132.1 tos 160
```

```
traceroute to 11.19.142.1 (11.19.142.1) from 11.19.132.1, 30 hops max, 40 byte packets
```

```
1 10.0.0.2 (10.0.0.2) 1.135 ms !N 1.007 ms !N 0.954 ms !N
```

Related Documentation

- [Example: Configuring Multitopology Routing Based on Applications on page 13](#)

CHAPTER 4

Multitopology Routing Configuration Statements

community (Protocols BGP)

| | |
|---------------------------------|---|
| Syntax | <pre>community { target <i>identifier</i>; }</pre> |
| Hierarchy Level | <pre>[edit logical-systems <i>logical-system-name</i> protocols bgp family (inet inet6) unicast topology <i>name</i>], [edit logical-systems <i>logical-system-name</i> protocols bgp group <i>group-name</i> family (inet inet6) unicast topology <i>name</i>], [edit logical-systems <i>logical-system-name</i> protocols bgp group <i>group-name</i> neighbor <i>address</i> family (inet inet6) unicast topology <i>name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols bgp family (inet inet6) unicast topology <i>name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols bgp group <i>group-name</i> family (inet inet6) unicast topology <i>name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols bgp group <i>group-name</i> neighbor <i>address</i> family (inet inet6) unicast topology <i>name</i>], [edit protocols bgp family (inet inet6) unicast topology <i>name</i>], [edit protocols bgp group <i>group-name</i> family (inet inet6) unicast topology <i>name</i>], [edit protocols bgp group <i>group-name</i> neighbor <i>address</i> family (inet inet6) unicast topology <i>name</i>], [edit routing-instances <i>routing-instance-name</i> protocols bgp family (inet inet6) unicast topology <i>name</i>], [edit routing-instances <i>routing-instance-name</i> protocols bgp group <i>group-name</i> family (inet inet6) unicast topology <i>name</i>], [edit routing-instances <i>routing-instance-name</i> protocols bgp group <i>group-name</i> neighbor <i>address</i> family (inet inet6) topology <i>name</i>]</pre> |
| Release Information | Statement introduced in Junos OS Release 9.0. |
| Description | Configure the community to identify the multitopology routes. BGP uses the target community identifier to install the routes it learns in the appropriate multitopology routing tables. |
| Options | target <i>identifier</i> —Configure the destination to which the route is going. |
| Required Privilege Level | routing—To view this statement in the configuration. routing-control—To add this statement to the configuration. |
| Related Documentation | <ul style="list-style-type: none">• Example: Configuring Multitopology Routing Based on Applications on page 13• Example: Configuring Origin Validation for BGP |


rib (Multitopology Routing)

| | |
|---------------------------------|---|
| Syntax | <pre> rib <i>routing-table-name</i> { static { route <i>destination-prefix</i> { <i>next-hop</i>; } <i>static-options</i>; } } </pre> |
| Hierarchy Level | <p>[edit logical-systems <i>logical-system-name</i> routing-options], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> routing-options], [edit routing-instances <i>routing-instance-name</i> routing-options], [edit routing-options]</p> |
| Release Information | Statement support for multitopology routing introduced in Junos OS Release 9.0. Statement introduced in Junos OS Release 9.0 for EX Series switches. |
| Description | Configure a static route to install routes in the routing table for a specific topology. |
| Options | <p><i>routing-table-name</i>—Name of the routing table for a topology. Use the following format: <i>logical-system-name/routing-instance-name:topology-name.protocol.identifier</i>. Include the routing instance string only if the instance is not the master. The logical system string is included only if the logical system identifier has a value other than 0 (zero). Each routing table for a topology includes a colon (:) before the topology name. <i>protocol</i> is the protocol family, which can be inet or inet6. <i>identifier</i> is the positive integer that specifies the instance of the routing table. For example, to install IPv6 routes to the routing table for a topology named voice in the master instance, include :voice.inet6.0.</p> <p>The remaining statements are explained separately.</p> |
| Required Privilege Level | <p>routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.</p> |
| Related Documentation | <ul style="list-style-type: none"> • Examples: Configuring Static Routes • Examples: Configuring Multiprotocol BGP • Example: Configuring Martian Addresses • static |

topologies (Multitopology Routing)

| | |
|---------------------------------|---|
| Syntax | <pre>topologies { family (inet inet6) { topology <i>topology-name</i>; } }</pre> |
| Hierarchy Level | [edit logical-systems <i>logical-system-name</i> routing-options], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> routing-options], [edit routing-instances <i>routing-instance-name</i> routing-options], [edit routing-options] |
| Release Information | Statement introduced in Junos OS Release 9.0. Statement introduced in Junos OS Release 12.3 for ACX Series routers. |
| Description | Configure a topology for multitopology routing. Each topology creates a new routing table that is populated with direct routes from the topology. |
| Options | family —Configure the type of family address type. inet —IPv4 inet6 —IPv6 The remaining statement is explained separately. |
| Required Privilege Level | routing—To view this statement in the configuration. routing-control—To add this statement to the configuration. |
| Related Documentation | <ul style="list-style-type: none">• Example: Configuring Multitopology Routing Based on Applications on page 13• Example: Configuring Multitopology Routing Based on a Multicast Source on page 39 |

topology (Filter-Based Forwarding)

| | |
|---------------------------------|--|
| Syntax | <code>topology <i>topology-name</i>;</code> |
| Hierarchy Level | <p>[edit firewall family (inet inet6) filter <i>filter-name</i> term <i>term-name</i> then],</p> <p>[edit firewall family (inet inet6) filter <i>filter-name</i> term <i>term-name</i> then logical-system <i>logical-system-name</i>],</p> <p>[edit firewall family (inet inet6) filter <i>filter-name</i> term <i>term-name</i> then logical-system <i>logical-system-name</i> routing-instance <i>routing-instance-name</i>],</p> <p>[edit firewall family (inet inet6) filter <i>filter-name</i> term <i>term-name</i> then routing-instance <i>routing-instance-name</i>]</p> |
| Release Information | Statement introduced in Junos OS Release 9.0. |
| Description | <p>Configure a topology for filter-based forwarding for multitopology routing. The firewall filter you apply to the ingress interface is used to look up traffic against the configured topology, and, if a route matches the conditions you configure for the term, the route is accepted and added to the routing table for the specific topology.</p> <p>There are multiple ways to configure a topology for filter-based forwarding, depending on the type of instance or logical system you want to specify for the forwarding class.</p> |
| | <div>  <p>NOTE: The options for logical system and routing instance precede the topology statement with the then statement.</p> </div> |
| Options | <i>topology-name</i> —Name of a topology against which you want to match traffic. |
| Required Privilege Level | <p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p> |
| Related Documentation | <ul style="list-style-type: none"> • Example: Configuring Multitopology Routing Based on Applications on page 13 • Routing Policy Configuration Guide |

topology (Multitopology Routing)

| | |
|---------------------------------|---|
| Syntax | <code>topology <i>topology-name</i>;</code> |
| Hierarchy Level | <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> routing-options topologies family (inet inet6)],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-options topologies family (inet inet6)],</code> <code>[edit routing-instances <i>routing-instance-name</i> routing-options topologies family (inet inet6)],</code> <code>[edit routing-options topologies family (inet inet6)]</code> |
| Release Information | Statement introduced in Junos OS Release 9.0. Statement introduced in Junos OS Release 12.3 for ACX Series routers. |
| Description | Configure the name of a topology configured to run multitopology routing. |
| Options | <i>topology-name</i> —Name of the topology. Include a string value that describes the type of traffic, such as voice or video. For IPv4 multicast traffic, include ipv4-multicast as the name. |
| Required Privilege Level | <code>routing</code> —To view this statement in the configuration. <code>routing-control</code> —To add this statement to the configuration. |
| Related Documentation | <ul style="list-style-type: none">• Example: Configuring Multitopology Routing Based on Applications on page 13• Example: Configuring Multitopology Routing Based on a Multicast Source on page 39 |

topology (OSPF)

| | |
|---------------------------------|--|
| Syntax | <pre> topology (default ipv4-multicast <i>name</i>) { spf-options { delay <i>milliseconds</i>; holddown <i>milliseconds</i>; rapid-runs <i>number</i>; } topology-id <i>number</i>; } </pre> |
| Hierarchy Level | <p>[edit logical-systems <i>logical-system-name</i> protocols ospf], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols ospf], [edit protocols ospf], [edit routing-instances <i>routing-instance-name</i> protocols ospf]</p> |
| Release Information | <p>Statement introduced in Junos OS Release 9.0. Statement introduced in Junos OS Release 11.3 for the QFX Series.</p> |
| Description | <p>Enable a topology for OSPF multitopology routing. You must first configure one or more topologies under the [edit routing-options] hierarchy level.</p> |
| Options | <p>default—Name of the default topology. This topology is automatically created, and all routes that correspond to it are automatically added to the inet.0 routing table. You can modify certain default parameters, such as for the SPF algorithm.</p> <p>ipv4-multicast—Name of the topology for IPv4 multicast traffic.</p> <p><i>name</i>—Name of a topology you configured at the [edit routing-options] hierarchy level to create a topology for a specific type of traffic, such as voice or video.</p> <p>The remaining statements are explained separately.</p> |
| Required Privilege Level | <p>routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.</p> |
| Related Documentation | <ul style="list-style-type: none"> • Example: Configuring Multitopology Routing Based on Applications on page 13 • Example: Configuring Multitopology Routing Based on a Multicast Source on page 39 |

topology (OSPF Interface)

| | |
|---------------------------------|---|
| Syntax | <code>topology (ipv4-multicast <i>name</i>) { metric <i>metric</i>; }</code> |
| Hierarchy Level | [edit logical-systems <i>logical-system-name</i> protocols ospf area <i>area-id</i> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols ospf area <i>area-id</i> interface <i>interface-name</i>], [edit protocols ospf area <i>area-id</i> interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> protocols ospf area <i>area-id</i> interface <i>interface-name</i>] |
| Release Information | Statement introduced in Junos OS Release 9.0. |
| Description | <p>Configure interface-specific properties for multitopology OSPF, including topology-specific metric values for an interface.</p> <p>All OSPF 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. The default value for the OSPF metric for an interface is 1. You can modify the default value for an OSPF interface and configure a topology-specific metric for that interface. The topology-specific metric applies to routes advertised from the interface that belong only to that topology.</p> |
| Default | The default value of the topology metric is the same as the default metric value calculated by OSPF or the value configured for the OSPF metric. |
| Options | <p>ipv4-multicast—Name of the topology for IPv4 multicast traffic.</p> <p><i>name</i>—Name of a topology created under the [edit routing-options] hierarchy level.</p> <p>metric <i>metric</i>—Cost of a route from an OSPF interface. You can specify a metric value for a topology that is different from the value specified for the interface.</p> <p>Range: 1 through 65,535</p> <p>Default: 1</p> |
| Required Privilege Level | <p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p> |
| Related Documentation | <ul style="list-style-type: none">• Example: Configuring Multitopology Routing Based on Applications on page 13• Example: Configuring Multitopology Routing Based on a Multicast Source on page 39 |

topology-id

| | |
|---------------------------------|--|
| Syntax | <code>topology-id <i>number</i>;</code> |
| Hierarchy Level | <p>[edit logical-systems <i>logical-system-name</i> protocols ospf topology name],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols ospf topology name],</p> <p>[edit protocols ospf topology name],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols ospf topology name]</p> |
| Release Information | Statement introduced in Junos OS Release 9.0. |
| Description | Configure a topology identifier for a topology enabled for OSPF. |
| Default | The default identifier for the default topology is 0, and the default identifier for the topology for IPv4 multicast traffic is 1. These identifiers are predefined and cannot be modified. |
| Options | <p><i>number</i>—The integer value used to identify the topology.</p> <p>Range: 32 through 127</p> |
| Required Privilege Level | <p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p> |
| Related Documentation | <ul style="list-style-type: none"> • Example: Configuring Multitopology Routing Based on Applications on page 13 • Example: Configuring Multitopology Routing Based on a Multicast Source on page 39 • topology on page 73 |

PART 3

Administration

- [Multitopology Routing Operational Commands on page 79](#)

CHAPTER 5

Multitopology Routing Operational Commands

show ospf database

| | |
|---|--|
| Syntax | <pre>show ospf database <brief detail extensive summary> <advertising-router (address self)> <area area-id> <asbrsummary> <external> <instance instance-name> <link-local> <logical-system (all logical-system-name)> <lsa-id lsa-id> <netsummary> <network> <nssa> <opaque-area> <router></pre> |
| Syntax (EX Series Switches and QFX Series) | <pre>show ospf database <brief detail extensive summary> <advertising-router (address self)> <area area-id> <asbrsummary> <external> <instance instance-name> <link-local> <lsa-id lsa-id> <netsummary> <network> <nssa> <opaque-area> <router></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>advertising-router self (address self) option introduced in Junos OS Release 9.5.</p> <p>advertising-router self (address self) option introduced in Junos OS Release 9.5 for EX Series switches.</p> <p>Command introduced in Junos OS Release 11.3 for the QFX Series.</p> |
| Description | Display the entries in the OSPF version 2 (OSPFv2) link-state database, which contains data about link-state advertisement (LSA) packets. |
| Options | <p>none—Display standard information about entries in the OSPFv2 link-state database for all routing instances.</p> <p>brief detail extensive summary—(Optional) Display the specified level of output.</p> <p>advertising-router (address self)—(Optional) Display the LSAs advertised either by a particular routing device or by this routing device.</p> <p>area area-id—(Optional) Display the LSAs in a particular area.</p> |

asbrsummary—(Optional) Display summary AS boundary router LSA entries.

external—(Optional) Display external LSAs.

instance *instance-name*—(Optional) Display all OSPF database information under the named routing instance.

link-local—(Optional) Display information about link-local LSAs.

logical-system (all | *logical-system-name*)—(Optional) Perform this operation on all logical systems or on a particular logical system.

lsa-id *lsa-id*—(Optional) Display the LSA with the specified LSA identifier.

netsummary—(Optional) Display summary network LSAs.

network—(Optional) Display information about network LSAs.

nssa—(Optional) Display information about not-so-stubby area (NSSA) LSAs.

opaque-area—(Optional) Display opaque area-scope LSAs.

router—(Optional) Display information about router LSAs.

Required Privilege Level

view

Related Documentation

- [clear \(ospf | ospf3\) database](#)

List of Sample Output

[show ospf database on page 84](#)
[show ospf database brief on page 84](#)
[show ospf database detail on page 84](#)
[show ospf database extensive on page 85](#)
[show ospf database summary on page 88](#)

Output Fields

[Table 4 on page 81](#) describes the output fields for the **show ospf database** command. Output fields are listed in the approximate order in which they appear.

Table 4: show ospf database Output Fields

| Field Name | Field Description | Level of Output |
|----------------|--|-----------------|
| area | Area number. Area 0.0.0.0 is the backbone area. | All levels |
| Type | Type of link advertisement: ASBRSum , Extern , Network , NSSA , OpaqArea , Router , or Summary . | All levels |
| ID | LSA identifier included in the advertisement. An asterisk preceding the identifier marks database entries that originated from the local routing device. | All levels |
| Adv Rtr | Address of the routing device that sent the advertisement. | All levels |
| Seq | Link sequence number of the advertisement. | All levels |

Table 4: show ospf database Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|---|--|-------------------------|
| Age | Time elapsed since the LSA was originated, in seconds. | All levels |
| Opt | Optional OSPF capabilities associated with the LSA. | All levels |
| Cksum | Checksum value of the LSA. | All levels |
| Len | Length of the advertisement, in bytes. | All levels |
| Router | Router link-state advertisement information: <ul style="list-style-type: none"> bits—Flags describing the routing device that generated the LSP. link count—Number of links in the advertisement. id—ID of a routing device or subnet on the link. data—For stub networks, the subnet mask. Otherwise, the IP address of the routing device that generated the LSP. type—Type of link. It can be PointToPoint, Transit, Stub, or Virtual. TOS count—Number of type-of-service (ToS) entries in the advertisement. TOS 0 metric—Metric for ToS 0. TOS—Type-of-service (ToS) value. metric—Metric for the ToS. | detail extensive |
| Network | Network link-state advertisement information: <ul style="list-style-type: none"> mask—Network mask. attached router—ID of the attached neighbor. | detail extensive |
| Summary | Summary link-state advertisement information: <ul style="list-style-type: none"> mask—Network mask. TOS—Type-of-service (ToS) value. metric—Metric for the ToS. | detail extensive |
| Gen timer | How long until the LSA is regenerated. | extensive |
| Aging timer | How long until the LSA expires. | extensive |
| Installed <i>hh:mm:ss</i> ago | How long ago the route was installed. | extensive |
| expires in <i>hh:mm:ss</i> | How long until the route expires. | extensive |
| sent <i>hh:mm:ss</i> ago | How long ago the LSA was sent. | extensive |
| Last changed <i>hh:mm:ss</i> ago | How long ago the route was changed. | extensive |
| Change count | Number of times the route has changed. | extensive |

Table 4: show ospf database Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|---------------------|--|------------------|
| Ours | Indicates that this is a local advertisement. | extensive |
| Router LSAs | Number of router link-state advertisements in the link-state database. | summary |
| Network LSAs | Number of network link-state advertisements in the link-state database. | summary |
| Summary LSAs | Number of summary link-state advertisements in the link-state database. | summary |
| NSSA LSAs | Number of not-so-stubby area link-state advertisements in the link-state database. | summary |

Sample Output

show ospf database

```

user@host> show ospf database
OSPF link state database, Area 0.0.0.1
  Type      ID                Adv Rtr          Seq      Age  Opt  Cksum  Len
Router     10.255.70.103        10.255.70.103    0x80000002 215  0x20 0x4112 48
Router     *10.255.71.242        10.255.71.242    0x80000002 214  0x20 0x11b1 48
Summary    *23.1.1.0            10.255.71.242    0x80000002 172  0x20 0x6d72 28
Summary    *24.1.1.0            10.255.71.242    0x80000002 177  0x20 0x607e 28
NSSA       *33.1.1.1            10.255.71.242    0x80000002 217  0x28 0x73bd 36

      OSPF link state database, Area 0.0.0.2
  Type      ID                Adv Rtr          Seq      Age  Opt  Cksum  Len
Router     10.255.71.52        10.255.71.52     0x80000004 174  0x20 0xd021 36
Router     *10.255.71.242        10.255.71.242    0x80000003 173  0x20 0xe191 36
Network    *23.1.1.1            10.255.71.242    0x80000002 173  0x20 0x9c76 32
Summary    *12.1.1.0            10.255.71.242    0x80000001 217  0x20 0xfeec 28
Summary    *24.1.1.0            10.255.71.242    0x80000002 177  0x20 0x607e 28
NSSA       *33.1.1.1            10.255.71.242    0x80000001 222  0x28 0xe047 36

      OSPF link state database, Area 0.0.0.3
  Type      ID                Adv Rtr          Seq      Age  Opt  Cksum  Len
Router     10.255.71.238        10.255.71.238    0x80000003 179  0x20 0x3942 36
Router     *10.255.71.242        10.255.71.242    0x80000003 177  0x20 0xf37d 36
Network    *24.1.1.1            10.255.71.242    0x80000002 177  0x20 0xc591 32
Summary    *12.1.1.0            10.255.71.242    0x80000001 217  0x20 0xfeec 28
Summary    *23.1.1.0            10.255.71.242    0x80000002 172  0x20 0x6d72 28
NSSA       *33.1.1.1            10.255.71.242    0x80000001 222  0x28 0xeb3b 36

```

show ospf database brief

The output for the **show ospf database brief** command is identical to that for the **show ospf database** command. For sample output, see [show ospf database on page 84](#).

show ospf database detail

```

user@host> show ospf database detail
      OSPF link state database, Area 0.0.0.1
  Type      ID                Adv Rtr          Seq      Age  Opt  Cksum  Len
Router     10.255.70.103        10.255.70.103    0x80000002 261  0x20 0x4112 48
  bits 0x0, link count 2
  id 10.255.71.242, data 12.1.1.1, Type PointToPoint (1)
  TOS count 0, TOS 0 metric 1
  id 12.1.1.0, data 255.255.255.0, Type Stub (3)
  TOS count 0, TOS 0 metric 1
Router     *10.255.71.242        10.255.71.242    0x80000002 260  0x20 0x11b1 48
  bits 0x3, link count 2
  id 10.255.70.103, data 12.1.1.2, Type PointToPoint (1)
  TOS count 0, TOS 0 metric 1
  id 12.1.1.0, data 255.255.255.0, Type Stub (3)
  TOS count 0, TOS 0 metric 1
Summary    *23.1.1.0            10.255.71.242    0x80000002 218  0x20 0x6d72 28
  mask 255.255.255.0
  TOS 0x0, metric 1
Summary    *24.1.1.0            10.255.71.242    0x80000002 223  0x20 0x607e 28
  mask 255.255.255.0
  TOS 0x0, metric 1
NSSA       *33.1.1.1            10.255.71.242    0x80000002 263  0x28 0x73bd 36
  mask 255.255.255.255
  Type 2, TOS 0x0, metric 0, fwd addr 12.1.1.2, tag 0.0.0.0

      OSPF link state database, Area 0.0.0.2

```

```

Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router  10.255.71.52    10.255.71.52  0x80000004 220  0x20 0xd021 36
  bits 0x0, link count 1
  id 23.1.1.1, data 23.1.1.2, Type Transit (2)
  TOS count 0, TOS 0 metric 1
Router *10.255.71.242 10.255.71.242 0x80000003 219  0x20 0xe191 36
  bits 0x3, link count 1
  id 23.1.1.1, data 23.1.1.1, Type Transit (2)
  TOS count 0, TOS 0 metric 1
Network *23.1.1.1    10.255.71.242 0x80000002 219  0x20 0x9c76 32
  mask 255.255.255.0
  attached router 10.255.71.242
  attached router 10.255.71.52
Summary *12.1.1.0    10.255.71.242 0x80000001 263  0x20 0xfeec 28
  mask 255.255.255.0
  TOS 0x0, metric 1
Summary *24.1.1.0    10.255.71.242 0x80000002 223  0x20 0x607e 28
  mask 255.255.255.0
  TOS 0x0, metric 1
NSSA   *33.1.1.1      10.255.71.242 0x80000001 268  0x28 0xe047 36
  mask 255.255.255.255
  Type 2, TOS 0x0, metric 0, fwd addr 23.1.1.1, tag 0.0.0.0

```

OSPF link state database, Area 0.0.0.3

```

Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router  10.255.71.238 10.255.71.238 0x80000003 225  0x20 0x3942 36
  bits 0x0, link count 1
  id 24.1.1.1, data 24.1.1.2, Type Transit (2)
  TOS count 0, TOS 0 metric 1
Router *10.255.71.242 10.255.71.242 0x80000003 223  0x20 0xf37d 36
  bits 0x3, link count 1
  id 24.1.1.1, data 24.1.1.1, Type Transit (2)
  TOS count 0, TOS 0 metric 1
Network *24.1.1.1    10.255.71.242 0x80000002 223  0x20 0xc591 32
  mask 255.255.255.0
  attached router 10.255.71.242
  attached router 10.255.71.238
Summary *12.1.1.0    10.255.71.242 0x80000001 263  0x20 0xfeec 28
  mask 255.255.255.0
  TOS 0x0, metric 1
Summary *23.1.1.0    10.255.71.242 0x80000002 218  0x20 0x6d72 28
  mask 255.255.255.0
  TOS 0x0, metric 1
NSSA   *33.1.1.1      10.255.71.242 0x80000001 268  0x28 0xeb3b 36
  mask 255.255.255.255
  Type 2, TOS 0x0, metric 0, fwd addr 24.1.1.1, tag 0.0.0.0

```

show ospf database extensive

```

user@host> show ospf database extensive
  OSPF link state database, Area 0.0.0.1
Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router  10.255.70.103 10.255.70.103 0x80000002 286  0x20 0x4112 48
  bits 0x0, link count 2
  id 10.255.71.242, data 12.1.1.1, Type PointToPoint (1)
  TOS count 0, TOS 0 metric 1
  id 12.1.1.0, data 255.255.255.0, Type Stub (3)
  TOS count 0, TOS 0 metric 1
  Aging timer 00:55:14
  Installed 00:04:43 ago, expires in 00:55:14
  Last changed 00:04:43 ago, Change count: 2
Router *10.255.71.242 10.255.71.242 0x80000002 285  0x20 0x11b1 48
  bits 0x3, link count 2

```

```

id 10.255.70.103, data 12.1.1.2, Type PointToPoint (1)
TOS count 0, TOS 0 metric 1
id 12.1.1.0, data 255.255.255.0, Type Stub (3)
TOS count 0, TOS 0 metric 1
Gen timer 00:45:15
Aging timer 00:55:15
Installed 00:04:45 ago, expires in 00:55:15, sent 00:04:43 ago
Last changed 00:04:45 ago, Change count: 2, Ours
Summary *23.1.1.0          10.255.71.242    0x80000002    243    0x20 0x6d72    28
mask 255.255.255.0
TOS 0x0, metric 1
Gen timer 00:45:57
Aging timer 00:55:57
Installed 00:04:03 ago, expires in 00:55:57, sent 00:04:01 ago
Last changed 00:04:48 ago, Change count: 1, Ours
Summary *24.1.1.0          10.255.71.242    0x80000002    248    0x20 0x607e    28
mask 255.255.255.0
TOS 0x0, metric 1
Gen timer 00:45:52
Aging timer 00:55:52
Installed 00:04:08 ago, expires in 00:55:52, sent 00:04:06 ago
Last changed 00:04:48 ago, Change count: 1, Ours
NSSA *33.1.1.1            10.255.71.242    0x80000002    288    0x28 0x73bd    36
mask 255.255.255.255
Type 2, TOS 0x0, metric 0, fwd addr 12.1.1.2, tag 0.0.0.0
Gen timer 00:45:12
Aging timer 00:55:12
Installed 00:04:48 ago, expires in 00:55:12, sent 00:04:48 ago
Last changed 00:04:48 ago, Change count: 2, Ours

OSPF link state database, Area 0.0.0.2
Type      ID          Adv Rtr      Seq      Age  Opt  Cksum  Len
Router 10.255.71.52    10.255.71.52 0x80000004  245  0x20 0xd021  36
bits 0x0, link count 1
id 23.1.1.1, data 23.1.1.2, Type Transit (2)
TOS count 0, TOS 0 metric 1
Aging timer 00:55:55
Installed 00:04:02 ago, expires in 00:55:55
Last changed 00:04:02 ago, Change count: 2
Router *10.255.71.242    10.255.71.242 0x80000003  244  0x20 0xe191  36
bits 0x3, link count 1
id 23.1.1.1, data 23.1.1.1, Type Transit (2)
TOS count 0, TOS 0 metric 1
Gen timer 00:45:56
Aging timer 00:55:56
Installed 00:04:04 ago, expires in 00:55:56, sent 00:04:02 ago
Last changed 00:04:04 ago, Change count: 2, Ours
Network *23.1.1.1        10.255.71.242 0x80000002  244  0x20 0x9c76  32
mask 255.255.255.0
attached router 10.255.71.242
attached router 10.255.71.52
Gen timer 00:45:56
Aging timer 00:55:56
Installed 00:04:04 ago, expires in 00:55:56, sent 00:04:02 ago
Last changed 00:04:04 ago, Change count: 1, Ours
Summary *12.1.1.0        10.255.71.242 0x80000001  288  0x20 0xfeec  28
mask 255.255.255.0
TOS 0x0, metric 1
Gen timer 00:45:12
Aging timer 00:55:12
Installed 00:04:48 ago, expires in 00:55:12, sent 00:04:04 ago

```

```

Last changed 00:04:48 ago, Change count: 1, Ours
Summary *24.1.1.0      10.255.71.242    0x80000002    248    0x20 0x607e    28
mask 255.255.255.0
TOS 0x0, metric 1
Gen timer 00:45:52
Aging timer 00:55:52
Installed 00:04:08 ago, expires in 00:55:52, sent 00:04:04 ago
Last changed 00:04:48 ago, Change count: 1, Ours
NSSA *33.1.1.1      10.255.71.242    0x80000001    293    0x28 0xe047    36
mask 255.255.255.255
Type 2, TOS 0x0, metric 0, fwd addr 23.1.1.1, tag 0.0.0.0
Gen timer 00:45:07
Aging timer 00:55:07
Installed 00:04:53 ago, expires in 00:55:07, sent 00:04:04 ago
Last changed 00:04:53 ago, Change count: 1, Ours

```

```

OSPF link state database, Area 0.0.0.3
Type      ID      Adv Rtr      Seq      Age  Opt  Cksum  Len
Router  10.255.71.238  10.255.71.238  0x80000003  250  0x20 0x3942  36
bits 0x0, link count 1
id 24.1.1.1, data 24.1.1.2, Type Transit (2)
TOS count 0, TOS 0 metric 1
Aging timer 00:55:50
Installed 00:04:07 ago, expires in 00:55:50
Last changed 00:04:07 ago, Change count: 2
Router *10.255.71.242  10.255.71.242  0x80000003  248  0x20 0xf37d  36
bits 0x3, link count 1
id 24.1.1.1, data 24.1.1.1, Type Transit (2)
TOS count 0, TOS 0 metric 1
Gen timer 00:45:52
Aging timer 00:55:52
Installed 00:04:08 ago, expires in 00:55:52, sent 00:04:06 ago
Last changed 00:04:08 ago, Change count: 2, Ours
Network *24.1.1.1      10.255.71.242    0x80000002    248    0x20 0xc591    32
mask 255.255.255.0
attached router 10.255.71.242
attached router 10.255.71.238
Gen timer 00:45:52
Aging timer 00:55:52
Installed 00:04:08 ago, expires in 00:55:52, sent 00:04:06 ago
Last changed 00:04:08 ago, Change count: 1, Ours
Summary *12.1.1.0      10.255.71.242    0x80000001    288    0x20 0xfeec    28
mask 255.255.255.0
TOS 0x0, metric 1
Gen timer 00:45:12
Aging timer 00:55:12
Installed 00:04:48 ago, expires in 00:55:12, sent 00:04:13 ago
Last changed 00:04:48 ago, Change count: 1, Ours
Summary *23.1.1.0      10.255.71.242    0x80000002    243    0x20 0x6d72    28
mask 255.255.255.0
TOS 0x0, metric 1
Gen timer 00:45:57
Aging timer 00:55:57
Installed 00:04:03 ago, expires in 00:55:57, sent 00:04:01 ago
Last changed 00:04:48 ago, Change count: 1, Ours
NSSA *33.1.1.1      10.255.71.242    0x80000001    293    0x28 0xeb3b    36
mask 255.255.255.255
Type 2, TOS 0x0, metric 0, fwd addr 24.1.1.1, tag 0.0.0.0
Gen timer 00:45:07
Aging timer 00:55:07
Installed 00:04:53 ago, expires in 00:55:07, sent 00:04:13 ago

```

Last changed 00:04:53 ago, Change count: 1, Ours

**show ospf database
summary**

user@host> show ospf database summary

Area 0.0.0.1:
 2 Router LSAs
 2 Summary LSAs
 1 NSSA LSAs

Area 0.0.0.2:
 2 Router LSAs
 1 Network LSAs
 2 Summary LSAs
 1 NSSA LSAs

Area 0.0.0.3:
 2 Router LSAs
 1 Network LSAs
 2 Summary LSAs
 1 NSSA LSAs

Externals:

Interface fe-2/2/1.0:

Interface ge-0/3/2.0:

Interface so-0/1/2.0:

Interface so-0/1/2.0:

show ospf3 database

| | |
|---|--|
| Syntax | <pre>show ospf3 database <brief detail extensive summary> <advertising-router (address self)> <area area-id> <external> <instance instance-name> <inter-area-prefix> <inter-area-router> <intra-area-prefix> <link> <link-local> <logical-system (all logical-system-name)> <lsa-id lsa-id> <network> <nssa> <realm (ipv4-multicast ipv4-unicast ipv6-multicast)> <router></pre> |
| Syntax (EX Series Switches and QFX Series) | <pre>show ospf3 database <brief detail extensive summary> <advertising-router (address self)> <area area-id> <external> <instance instance-name> <inter-area-prefix> <inter-area-router> <intra-area-prefix> <link> <link-local> <lsa-id lsa-id> <network> <nssa> <router></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>realm option introduced in Junos OS Release 9.2.</p> <p>advertising-router (address self) option introduced in Junos Release 9.5.</p> <p>advertising-router (address self) option introduced in Junos OS Release 9.5 for EX Series switches.</p> <p>Command introduced in Junos OS Release 11.3 for the QFX Series.</p> |
| Description | Display the entries in the OSPF version 3 (OSPFv3) link-state database, which contains data about link-state advertisement (LSA) packets. |
| Options | <p>none—Display standard information about all entries in the OSPFv3 link-state database.</p> <p>brief detail extensive summary—(Optional) Display the specified level of output.</p> <p>advertising-router (address self)—(Optional) Display the LSAs advertised either by a particular routing device or by this routing device.</p> |

area *area-id*—(Optional) Display the LSAs in a particular area.

external—(Optional) Display external LSAs.

instance *instance-name*—(Optional) Display all OSPF database information under the named routing instance.

inter-area-prefix—(Optional) Display information about interarea-prefix LSAs.

inter-area-router—(Optional) Display information about interarea-router LSAs.

intra-area-prefix—(Optional) Display information about intra-area-prefix LSAs.

link—(Optional) Display information about link LSAs.

link-local—(Optional) Display information about link-local LSAs.

logical-system (**all** | *logical-system-name*)—(Optional) Perform this operation on all logical systems or on a particular logical system.

lsa-id *lsa-id*—(Optional) Display the LSA with the specified LSA identifier.

network—(Optional) Display information about network LSAs.

nssa—(Optional) Display information about not-so-stubby area (NSSA) LSAs.

realm (**ipv4-multicast** | **ipv4-unicast** | **ipv6-multicast**)—(Optional) Display information about the specified OSPFv3 realm, or address family. Use the **realm** option to specify an address family other than IPv6 unicast, which is the default.

router—(Optional) Display information about router LSAs.

Required Privilege Level

view

Related Documentation

- [clear \(ospf | ospf3\) database](#)

List of Sample Output

[show ospf3 database brief on page 96](#)
[show ospf3 database extensive on page 96](#)
[show ospf3 database summary on page 99](#)

Output Fields

[Table 5 on page 90](#) lists the output fields for the **show ospf3 database** command. Output fields are listed in the approximate order in which they appear.

Table 5: show ospf3 database Output Fields

| Field Name | Field Description | Level of Output |
|---|---|------------------------|
| OSPF link state database, area <i>area-number</i> | Entries in the link-state database for this area. | brief detail extensive |
| OSPF AS SCOPE link state database | Entries in the AS scope link-state database. | brief detail extensive |

Table 5: show ospf3 database Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|--|--|------------------------|
| OSPF Link-Local link state database, interface <i>interface-name</i> | Entries in the link-local link-state database for this interface. | brief detail extensive |
| area | Area number. Area 0.0.0.0 is the backbone area. | All levels |
| Type | Type of link advertisement: Extern , InterArPfx , InterArRtr , IntraArPrx , Link , Network , NSSA , or Router . | brief detail extensive |
| ID | Link identifier included in the advertisement. An asterisk (*) preceding the identifier marks database entries that originated from the local routing device. | brief detail extensive |
| Adv Rtr | Address of the routing device that sent the advertisement. | brief detail extensive |
| Seq | Link sequence number of the advertisement. | brief detail extensive |
| Age | Time elapsed since the LSA was originated, in seconds. | brief detail extensive |
| Cksum | Checksum value of the LSA. | brief detail extensive |
| Len | Length of the advertisement, in bytes. | brief detail extensive |
| Router (Router Link-State Advertisements) | | |
| bits | Flags describing the routing device that generated the LSP. | detail extensive |
| Options | Option bits carried in the router LSA. | detail extensive |
| For Each Router Link | | |
| Type | Type of interface. The value of all other output fields describing a routing device interface depends on the interface's type: <ul style="list-style-type: none"> • PointToPoint (1)—Point-to-point connection to another routing device. • Transit (2)—Connection to a transit network. • Virtual (4)—Virtual link. | detail extensive |
| Loc-if-id | Local interface ID assigned to the interface that uniquely identifies the interface with the routing device. | detail extensive |
| Nbr-if-id | Interface ID of the neighbor's interface for this routing device link. | detail extensive |
| Nbr-rtr-id | Router ID of the neighbor routing device (for type 2 interfaces, the attached link's designated router). | detail extensive |
| Metric | Cost of the router link. | detail extensive |
| Gen timer | How long until the LSA is regenerated, in the format <i>hours:minutes:seconds</i> . | extensive |

Table 5: show ospf3 database Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|--|--|------------------|
| Aging timer | How long until the LSA expires, in the format <i>hours:minutes:seconds</i> . | extensive |
| Installed <i>nn:nn:nn</i> ago | How long ago the route was installed, in the format <i>hours:minutes:seconds</i> . | extensive |
| expires in <i>nn:nn:nn</i> | How long until the route expires, in the format <i>hours:minutes:seconds</i> . | extensive |
| sent <i>nn:nn:nn</i> ago | Time elapsed since the LSA was last transmitted or flooded to an adjacency or an interface, respectively, in the format <i>hours:minutes:seconds</i> . | extensive |
| Ours | Indicates that this is a local advertisement. | extensive |
| Network (Network Link-State Advertisements) | | |
| Options | Option bits carried in the network LSA. | detail extensive |
| Attached Router | Router IDs of each of the routing devices attached to the link. Only routing devices that are fully adjacent to the designated router are listed. The designated router includes itself in this list. | detail extensive |
| InterArPfx (Interarea-Prefix Link-State Advertisements) | | |
| Prefix | IPv6 address prefix. | detail extensive |
| Prefix-options | Option bit associated with the prefix. | detail extensive |
| Metric | Cost of this route. Expressed in the same units as the interface costs in the router LSAs. When the interarea-prefix LSA is describing a route to a range of addresses, the cost is set to the maximum cost to any reachable component of the address range. | detail extensive |
| Gen timer | How long until the LSA is regenerated, in the format <i>hours:minutes:seconds</i> . | extensive |
| Aging timer | How long until the LSA expires, in the format <i>hours:minutes:seconds</i> . | extensive |
| Installed <i>nn:nn:nn</i> ago | How long ago the route was installed, in the format <i>hours:minutes:seconds</i> . | extensive |
| expires in <i>nn:nn:nn</i> | How long until the route expires, in the format <i>hours:minutes:seconds</i> . | extensive |
| sent <i>nn:nn:nn</i> ago | Time elapsed since the LSA was last transmitted or flooded to an adjacency or an interface, respectively, in the format <i>hours:minutes:seconds</i> . | extensive |
| Ours | Indicates that this is a local advertisement. | extensive |
| InterArRtr (Interarea-Router Link-State Advertisements) | | |
| Dest-router-id | Router ID of the routing device described by the LSA. | detail extensive |
| options | Optional capabilities supported by the routing device. | detail extensive |

Table 5: show ospf3 database Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|--|--|-------------------------|
| Metric | Cost of this route. Expressed in the same units as the interface costs in the router LSAs. When the interarea-prefix LSA is describing a route to a range of addresses, the cost is set to the maximum cost to any reachable component of the address range. | detail extensive |
| Prefix | IPv6 address prefix. | extensive |
| Prefix-options | Option bit associated with the prefix. | extensive |
| Extern (External Link-State Advertisements) | | |
| Prefix | IPv6 address prefix. | detail extensive |
| Prefix-options | Option bit associated with the prefix. | detail extensive |
| Metric | Cost of the route, which depends on the value of Type . | detail extensive |
| Type <i>n</i> | Type of external metric: Type 1 or Type 2 . | detail extensive |
| Aging timer | How long until the LSA expires, in the format <i>hours:minutes:seconds</i> . | extensive |
| Installed <i>nn:nn:nn</i> ago | How long ago the route was installed, in the format <i>hours:minutes:seconds</i> . | extensive |
| expires in <i>nn:nn:nn</i> | How long until the route expires, in the format <i>hours:minutes:seconds</i> . | extensive |
| sent <i>nn:nn:nn</i> ago | Time elapsed since the LSA was last transmitted or flooded to an adjacency or an interface, respectively, in the format <i>hours:minutes:seconds</i> . | extensive |
| Link (Link-State Advertisements) | | |
| IPv6-Address | IPv6 link-local address on the link for which this link LSA originated. | detail extensive |
| Options | Option bits carried in the link LSA. | detail extensive |
| priority | Router priority of the interface attaching the originating routing device to the link. | detail extensive |
| Prefix-count | Number of IPv6 address prefixes contained in the LSA. The rest of the link LSA contains a list of IPv6 prefixes to be associated with the link. | detail extensive |
| Prefix | IPv6 address prefix. | detail extensive |
| Prefix-options | Option bit associated with the prefix. | detail extensive |
| Gen timer | How long until the LSA is regenerated, in the format <i>hours:minutes:seconds</i> . | extensive |
| Aging timer | How long until the LSA expires, in the format <i>hours:minutes:seconds</i> . | extensive |

Table 5: show ospf3 database Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|---|---|------------------|
| Installed <i>nn:nn:nn</i> ago | How long ago the route was installed, in the format <i>hours:minutes:seconds</i> . | extensive |
| expires in <i>nn:nn:nn</i> | How long until the route expires, in the format <i>hours:minutes:seconds</i> . | extensive |
| sent <i>nn:nn:nn</i> ago | Time elapsed since the LSA was last transmitted or flooded to an adjacency or an interface, respectively, in the format <i>hours:minutes:seconds</i> . | extensive |
| Ours | Indicates that this is a local advertisement. | extensive |
| IntraArPfx (Intra-Area-Prefix Link-State Advertisements) | | |
| Ref-lsa-type | LSA type of the referenced LSA. <ul style="list-style-type: none"> Router—Address prefixes are associated with a router LSA. Network—Address prefixes are associated with a network LSA. | detail extensive |
| Ref-lsa-id | Link-state ID of the referenced LSA. | detail extensive |
| Ref-router-id | Advertising router ID of the referenced LSA. | detail extensive |
| Prefix-count | Number of IPv6 address prefixes contained in the LSA. The rest of the link LSA contains a list of IPv6 prefixes to be associated with the link. | detail extensive |
| Prefix | IPv6 address prefix. | detail extensive |
| Prefix-options | Option bit associated with the prefix. | detail extensive |
| Metric | Cost of this prefix. Expressed in the same units as the interface costs in the router LSAs. | detail extensive |
| Gen timer | How long until the LSA is regenerated, in the format <i>hours:minutes:seconds</i> . | extensive |
| Aging timer | How long until the LSA expires, in the format <i>hours:minutes:seconds</i> . | extensive |
| Installed <i>hh:mm:ss</i> ago | How long ago the route was installed, in the format <i>hours:minutes:seconds</i> . | extensive |
| expires in <i>hh:mm:ss</i> | How long until the route expires, in the format <i>hours:minutes:seconds</i> . | extensive |
| sent <i>hh:mm:ss</i> ago | Time elapsed since the LSA was last transmitted or flooded to an adjacency or an interface, respectively, in the format <i>hours:minutes:seconds</i> . | extensive |
| <i>n</i> Router LSAs | Number of router LSAs in the link-state database. | summary |
| <i>n</i> Network LSAs | Number of network LSAs in the link-state database. | summary |
| <i>n</i> InterArPfx LSAs | Number of interarea-prefix LSAs in the link-state database. | summary |

Table 5: show ospf3 database Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|------------------------------------|--|-----------------|
| <i>n</i> InterArRtr LSAs | Number of interarea-router LSAs in the link-state database. | summary |
| <i>n</i> IntraArPfx LSAs | Number of intra-area-prefix LSAs in the link-state database. | summary |
| Externals | Display of the external LSA database. | summary |
| <i>n</i> Extern LSAs | Number of external LSAs in the link-state database. | summary |
| Interface <i>interface-name</i> | Name of the interface for which link-local LSA information is displayed. | summary |
| <i>n</i> Link LSAs | Number of link LSAs in the link-state database. | summary |

Sample Output

show ospf3 database brief

```

user@host> show ospf3 database brief
      OSPF3 link state database, area 0.0.0.0
      Type      ID          Adv Rtr      Seq          Age  Cksum  Len
Router        0.0.0.1      10.255.4.85  0x80000003   885  0xa697  40
Router        *0.0.0.1    10.255.4.93  0x80000002   953  0xc677  40
InterArPfx    *0.0.0.2    10.255.4.93  0x80000001   910  0xb96f  44
InterArRtr    *0.0.0.1    10.255.4.93  0x80000001   910  0xe159  32
IntraArPfx    *0.0.0.1    10.255.4.93  0x80000002   432  0x788f  72

      OSPF3 link state database, area 0.0.0.1
      Type      ID          Adv Rtr      Seq          Age  Cksum  Len
Router        *0.0.0.1    10.255.4.93  0x80000003   916  0xea40  40
Router        0.0.0.1    10.255.4.97  0x80000006   851  0xc95b  40
Network       0.0.0.2    10.255.4.97  0x80000002   916  0x4598  32
InterArPfx    *0.0.0.1    10.255.4.93  0x80000002   117  0xa980  44
InterArPfx    *0.0.0.2    10.255.4.93  0x80000002    62  0xd47e  44
NSSA          0.0.0.1    10.255.4.97  0x80000002   362  0x45ee  44
IntraArPfx    0.0.0.1    10.255.4.97  0x80000006   851  0x2f77  52

      OSPF3 AS SCOPE link state database
      Type      ID          Adv Rtr      Seq          Age  Cksum  Len
Extern        0.0.0.1    10.255.4.85  0x80000002    63  0x9b86  44
Extern        *0.0.0.1    10.255.4.93  0x80000001   910  0x59c9  44

      OSPF3 Link-Local link state database, interface ge-1/3/0.0
      Type      ID          Adv Rtr      Seq          Age  Cksum  Len
Link          *0.0.0.2    10.255.4.93  0x80000003   916  0x4dab  64

```

show ospf3 database extensive

```

user@host> show ospf3 database extensive
      OSPF3 link state database, area 0.0.0.0
      Type      ID          Adv Rtr      Seq          Age  Cksum  Len
Router        0.0.0.1      10.255.4.85  0x80000003  1028  0xa697  40
      bits 0x2, Options 0x13
      Type PointToPoint (1), Metric 10
      Loc-If-Id 2, Nbr-If-Id 3, Nbr-Rtr-Id 10.255.4.93
      Aging timer 00:42:51
      Installed 00:17:05 ago, expires in 00:42:52, sent 02:37:54 ago
Router        *0.0.0.1    10.255.4.93  0x80000002  1096  0xc677  40
      bits 0x3, Options 0x13
      Type PointToPoint (1), Metric 10
      Loc-If-Id 3, Nbr-If-Id 2, Nbr-Rtr-Id 10.255.4.85
      Gen timer 00:00:40
      Aging timer 00:41:44
      Installed 00:18:16 ago, expires in 00:41:44, sent 00:18:14 ago
      Ours
InterArPfx    *0.0.0.2    10.255.4.93  0x80000001  1053  0xb96f  44
      Prefix feee::10:10:2:0/126
      Prefix-options 0x0, Metric 10
      Gen timer 00:17:02
      Aging timer 00:42:26
      Installed 00:17:33 ago, expires in 00:42:27, sent 00:17:31 ago
      Ours
InterArPfx    *0.0.0.3    10.255.4.93  0x80000001  1053  0x71d3  44
      Prefix feee::10:255:4:97/128
      Prefix-options 0x0, Metric 10
      Gen timer 00:21:07
      Aging timer 00:42:26

```

```

    Installed 00:17:33 ago, expires in 00:42:27, sent 00:17:31 ago
    Ours
InterArRtr *0.0.0.1          10.255.4.93      0x80000001 1053 0xe159 32
  Dest-router-id 10.255.4.97, Options 0x19, Metric 10
  Gen timer 00:29:18
  Aging timer 00:42:26
  Installed 00:17:33 ago, expires in 00:42:27, sent 00:17:31 ago
  Ours
IntraArPfx 0.0.0.1          10.255.4.85      0x80000002 1028 0x2403 72
  Ref-lsa-type Router, Ref-lsa-id 0.0.0.0, Ref-router-id 10.255.4.85
  Prefix-count 2
  Prefix feee::10:255:4:85/128
    Prefix-options 0x2, Metric 0
  Prefix feee::10:10:1:0/126
    Prefix-options 0x0, Metric 10
  Aging timer 00:42:51
  Installed 00:17:05 ago, expires in 00:42:52, sent 02:37:54 ago
IntraArPfx *0.0.0.1          10.255.4.93      0x80000002 575 0x788f 72
  Ref-lsa-type Router, Ref-lsa-id 0.0.0.0, Ref-router-id 10.255.4.93
  Prefix-count 2
  Prefix feee::10:255:4:93/128
    Prefix-options 0x2, Metric 0
  Prefix feee::10:10:1:0/126
    Prefix-options 0x0, Metric 10
  Gen timer 00:33:23
  Aging timer 00:50:24
  Installed 00:09:35 ago, expires in 00:50:25, sent 00:09:33 ago
  OSPF3 link state database, area 0.0.0.1
Type      ID              Adv Rtr          Seq            Age  Cksum  Len
Router    *0.0.0.1          10.255.4.93     0x80000003     1059 0xea40 40
  bits 0x3, Options 0x19
  Type Transit (2), Metric 10
    Loc-If-Id 2, Nbr-If-Id 2, Nbr-Rtr-Id 10.255.4.97
  Gen timer 00:08:51
  Aging timer 00:42:20
  Installed 00:17:39 ago, expires in 00:42:21, sent 00:17:37 ago
Router     0.0.0.1          10.255.4.97     0x80000006     994 0xc95b 40
  bits 0x2, Options 0x19
  Type Transit (2), Metric 10
    Loc-If-Id 2, Nbr-If-Id 2, Nbr-Rtr-Id 10.255.4.97
  Aging timer 00:43:25
  Installed 00:16:31 ago, expires in 00:43:26, sent 02:37:54 ago
Network    0.0.0.2          10.255.4.97     0x80000002     1059 0x4598 32
  Options 0x11
  Attached router 10.255.4.97
  Attached router 10.255.4.93
  Aging timer 00:42:20
  Installed 00:17:36 ago, expires in 00:42:21, sent 02:37:54 ago
InterArPfx *0.0.0.1          10.255.4.93      0x80000002     260 0xa980 44
  Prefix feee::10:10:1:0/126
  Prefix-options 0x0, Metric 10
  Gen timer 00:45:39
  Aging timer 00:55:39
  Installed 00:04:20 ago, expires in 00:55:40, sent 00:04:18 ago
  Ours
InterArPfx *0.0.0.2          10.255.4.93      0x80000002     205 0xd47e 44
  Prefix feee::10:255:4:93/128
  Prefix-options 0x0, Metric 0
  Gen timer 00:46:35
  Aging timer 00:56:35
  Installed 00:03:25 ago, expires in 00:56:35, sent 00:03:23 ago

```

```

Ours
InterArPfx *0.0.0.3          10.255.4.93      0x80000001 1089 0x9bbb 44
Prefix feee::10:255:4:85/128
Prefix-options 0x0, Metric 10
Gen timer 00:04:46
Aging timer 00:41:51
Installed 00:18:09 ago, expires in 00:41:51, sent 00:17:43 ago
Ours
NSSA      0.0.0.1          10.255.4.97      0x80000002 505 0x45ee 44
Prefix feee::200:200:1:0/124
Prefix-options 0x8, Metric 10, Type 2,
Aging timer 00:51:35
Installed 00:08:22 ago, expires in 00:51:35, sent 02:37:54 ago
IntraArPfx 0.0.0.1          10.255.4.97      0x80000006 994 0x2f77 52
Ref-lsa-type Router, Ref-lsa-id 0.0.0.0, Ref-router-id 10.255.4.97
Prefix-count 1
Prefix feee::10:255:4:97/128
Prefix-options 0x2, Metric 0
Aging timer 00:43:25
Installed 00:16:31 ago, expires in 00:43:26, sent 02:37:54 ago
IntraArPfx 0.0.0.3          10.255.4.97      0x80000002 1059 0x4446 52
Ref-lsa-type Network, Ref-lsa-id 0.0.0.2, Ref-router-id 10.255.4.97
Prefix-count 1
Prefix feee::10:10:2:0/126
Prefix-options 0x0, Metric 0
Aging timer 00:42:20
Installed 00:17:36 ago, expires in 00:42:21, sent 02:37:54 ago
OSPF3 AS SCOPE link state database
Type      ID              Adv Rtr          Seq              Age  Cksum  Len
Extern    0.0.0.1              10.255.4.85      0x80000002       206  0x9b86 44
Prefix feee::100:100:1:0/124
Prefix-options 0x0, Metric 20, Type 2,
Aging timer 00:56:34
Installed 00:03:23 ago, expires in 00:56:34, sent 02:37:54 ago
Extern    *0.0.0.1              10.255.4.93      0x80000001       1053 0x59c9 44
Prefix feee::200:200:1:0/124
Prefix-options 0x0, Metric 10, Type 2,
Gen timer 00:25:12
Aging timer 00:42:26
Installed 00:17:33 ago, expires in 00:42:27, sent 00:17:31 ago

OSPF3 Link-Local link state database, interface ge-1/3/0.0
Type      ID              Adv Rtr          Seq              Age  Cksum  Len
Link      *0.0.0.2              10.255.4.93      0x80000003       1059 0x4dab 64
fe80::290:69ff:fe39:1cdb
Options 0x11, priority 128
Prefix-count 1
Prefix feee::10:10:2:0/126 Prefix-options 0x0
Gen timer 00:12:56
Aging timer 00:42:20
Installed 00:17:39 ago, expires in 00:42:21, sent 00:17:37 ago
Link      0.0.0.2              10.255.4.97      0x80000003       205  0xa87d 64
fe80::290:69ff:fe38:883e
Options 0x11, priority 128
Prefix-count 1
Prefix feee::10:10:2:0/126 Prefix-options 0x0
Aging timer 00:56:35
Installed 00:03:22 ago, expires in 00:56:35, sent 02:37:54 ago

OSPF3 Link-Local link state database, interface so-2/2/0.0
Type      ID              Adv Rtr          Seq              Age  Cksum  Len

```



```

Link          0.0.0.2          10.255.4.85          0x80000002    506 0x42bb 64
fe80::280:42ff:fe10:f169
Options 0x13, priority 128
Prefix-count 1
Prefix feee::10:10:1:0/126 Prefix-options 0x0
Aging timer 00:51:34
Installed 00:08:23 ago, expires in 00:51:34, sent 02:37:54 ago
Link          *0.0.0.3          10.255.4.93          0x80000002    505 0x6b7a 64
fe80::280:42ff:fe10:f177
Options 0x13, priority 128
Prefix-count 1
Prefix feee::10:10:1:0/126 Prefix-options 0x0
Gen timer 00:37:28
Aging timer 00:51:35
Installed 00:08:25 ago, expires in 00:51:35, sent 00:08:23 ago
Ours

```

show ospf3 database summary

```

user@host> show ospf3 database summary
Area 0.0.0.0:
  2 Router LSAs
  1 InterArPfx LSAs
  1 InterArRtr LSAs
  1 IntraArPfx LSAs
Area 0.0.0.1:
  2 Router LSAs
  1 Network LSAs
  2 InterArPfx LSAs
  1 NSSA LSAs
  1 IntraArPfx LSAs
Externals:
  2 Extern LSAs
Interface ge-1/3/0.0:
  1 Link LSAs
Interface lo0.0:
Interface so-2/2/0.0:
  1 Link LSAs

```

show (ospf | ospf3) interface

| | |
|---|---|
| Syntax | <code>show (ospf ospf3) interface</code> <code><brief detail extensive></code> <code><area <i>area-id</i>></code> <code><interface-name></code> <code><instance <i>instance-name</i>></code> <code><logical-system (all <i>logical-system-name</i>)></code> <code><realm (ipv4-multicast ipv4-unicast ipv6-multicast)></code> |
| Syntax (EX Series Switches and QFX Series) | <code>show (ospf ospf3) interface</code> <code><brief detail extensive></code> <code><area <i>area-id</i>></code> <code><interface-name></code> <code><instance <i>instance-name</i>></code> |
| Release Information | Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.0 for EX Series switches. area option introduced in Junos OS Release 9.2. area option introduced in Junos OS Release 9.2 for EX Series switches. realm option introduced in Junos OS Release 9.2. Command introduced in Junos OS Release 11.3 for the QFX Series. |
| Description | Display the status of OSPF interfaces. |
| Options | none —Display standard information about the status of all OSPF interfaces for all routing instances brief detail extensive —(Optional) Display the specified level of output. area <i>area-id</i> —(Optional) Display information about the interfaces that belong to the specified area. <i>interface-name</i> —(Optional) Display information for the specified interface. instance <i>instance-name</i> —(Optional) Display all OSPF interfaces under the named routing instance. logical-system (all <i>logical-system-name</i>) —(Optional) Perform this operation on all logical systems or on a particular logical system. realm (ipv4-multicast ipv4-unicast ipv6-multicast) —(OSPFv3 only) (Optional) Display information about the interfaces for the specified OSPFv3 realm, or address family. Use the realm option to specify an address family for OSPFv3 other than IPv6 unicast, which is the default. |
| Required Privilege Level | view |
| List of Sample Output | show ospf interface brief on page 103 show ospf interface detail on page 103 show ospf3 interface detail on page 103 |

[show ospf interface detail \(When Multiarea Adjacency Is Configured\) on page 103](#)
[show ospf interface area area-id on page 104](#)
[show ospf interface extensive \(When Flooding Reduction Is Enabled\) on page 104](#)
[show ospf interface extensive \(When LDP Synchronization Is Configured\) on page 105](#)

Output Fields Table 6 on page 101 lists the output fields for the **show (ospf | ospf3) interface** command. Output fields are listed in the approximate order in which they appear.

Table 6: show (ospf | ospf3) interface Output Fields

| Field Name | Field Description | Level of Output |
|-------------------------|---|-------------------------|
| Interface | Name of the interface running OSPF version 2 or OSPF version 3. | All levels |
| State | State of the interface: BDR , Down , DR , DRother , Loop , PtToPt , or Waiting . | All levels |
| Area | Number of the area that the interface is in. | All levels |
| DR ID | Address of the area's designated router. | All levels |
| BDR ID | Backup designated router for a particular subnet. | All levels |
| Nbrs | Number of neighbors on this interface. | All levels |
| Type | Type of interface: LAN , NBMA , P2MP , P2P , or Virtual . | detail extensive |
| Address | IP address of the neighbor. | detail extensive |
| Mask | Netmask of the neighbor. | detail extensive |
| Prefix-length | (OSPFv3) IPv6 prefix length, in bits. | detail extensive |
| OSPF3-Intf-Index | (OSPFv3) OSPF version 3 interface index. | detail extensive |
| MTU | Interface maximum transmission unit (MTU). | detail extensive |
| Cost | Interface cost (metric). | detail extensive |
| DR addr | Address of the designated router. | detail extensive |
| BDR addr | Address of the backup designated router. | detail extensive |
| Adj count | Number of adjacent neighbors. | detail extensive |
| Secondary | Indicates that this interface is configured as a secondary interface for this area. This interface can belong to more than one area, but can be designated as a primary interface for only one area. | detail extensive |
| Flood Reduction | Indicates that this interface is configured with flooding reduction. All self-originated LSAs from this interface are initially sent with the DoNotAge bit set. As a result, LSAs are refreshed only when a change occurs. | extensive |

Table 6: show (ospf | ospf3) interface Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|------------------------------|---|------------------|
| Priority | Router priority used in designated router (DR) election on this interface. | detail extensive |
| Flood list | List of link-state advertisements (LSAs) that might be about to flood this interface. | extensive |
| Ack list | Acknowledgment list. List of pending acknowledgments on this interface. | extensive |
| Descriptor list | List of packet descriptors. | extensive |
| Hello | Configured value for the hello timer. | detail extensive |
| Dead | Configured value for the dead timer. | detail extensive |
| Auth type | (OSPFv2) Authentication mechanism for sending and receiving OSPF protocol packets: <ul style="list-style-type: none"> • MD5—The MD5 mechanism is configured in accordance with RFC 2328. • None—No authentication method is configured. • Password—A simple password (RFC 2328) is configured. | detail extensive |
| Topology | (Multiarea adjacency) Name of topology: default or <i>name</i> . | |
| LDP sync state | (OSPFv2 and LDP synchronization) Current state of LDP synchronization: in sync , in holddown , and not supported . | extensive |
| reason | (OSPFv2 and LDP synchronization) Reason for the current state of LDP synchronization. The LDP session might be up or down, or adjacency might be up or down. | extensive |
| config holdtime | (OSPFv2 and LDP synchronization) Configured value of the hold timer. If the state is not synchronized, and the hold time is not infinity, the remaining field displays the number of seconds that remain until the configured hold timer expires. | extensive |
| IPSec SA name | (OSPFv2) Name of the IPSec security association name. | detail extensive |
| Active key ID | (OSPFv2 and MD5) Number from 0 to 255 that uniquely identifies an MD5 key. | detail extensive |
| Start time | (OSPFv2 and MD5) Time at which the routing device starts using an MD5 key to authenticate OSPF packets transmitted on the interface on which this key is configured. To authenticate received OSPF protocol packets, the key becomes effective immediately after the configuration is committed. If the start time option is not configured, the key is effective immediately for send and receive and is displayed as Start time 1970 Jan 01 00:00:00 PST . | detail extensive |
| ReXmit | Configured value for the Retransmit timer. | detail extensive |
| Stub, Not Stub, or Stub NSSA | Type of area. | detail extensive |

Sample Output

show ospf interface brief

```
user@host> show ospf interface brief
```

| Intf | State | Area | DR ID | BDR ID | Nbrs |
|------------|--------|---------|--------------|--------------|------|
| at-5/1/0.0 | PtToPt | 0.0.0.0 | 0.0.0.0 | 0.0.0.0 | 1 |
| ge-2/3/0.0 | DR | 0.0.0.0 | 192.168.4.16 | 192.168.4.15 | 1 |
| lo0.0 | DR | 0.0.0.0 | 192.168.4.16 | 0.0.0.0 | 0 |
| so-0/0/0.0 | Down | 0.0.0.0 | 0.0.0.0 | 0.0.0.0 | 0 |
| so-6/0/1.0 | PtToPt | 0.0.0.0 | 0.0.0.0 | 0.0.0.0 | 1 |
| so-6/0/2.0 | Down | 0.0.0.0 | 0.0.0.0 | 0.0.0.0 | 0 |
| so-6/0/3.0 | PtToPt | 0.0.0.0 | 0.0.0.0 | 0.0.0.0 | 1 |

show ospf interface detail

```
user@host> show ospf interface detail
```

| Interface | State | Area | DR ID | BDR ID | Nbrs |
|------------|-------|---------|---------------|----------------|------|
| fe-0/0/1.0 | BDR | 0.0.0.0 | 192.168.37.12 | 10.255.245.215 | 1 |

Type LAN, address 192.168.37.11, Mask 255.255.255.248, MTU 4460, Cost 40
DR addr 192.168.37.12, BDR addr 192.168.37.11, Adj count 1, Priority 128
Hello 10, Dead 40, ReXmit 5, Not Stub

| Interface | State | Area | DR ID | BDR ID | Nbrs |
|------------|--------|---------|---------|---------|------|
| tl-0/2/1.0 | PtToPt | 0.0.0.0 | 0.0.0.0 | 0.0.0.0 | 0 |

Type P2P, Address 0.0.0.0, Mask 0.0.0.0, MTU 1500, Cost 2604
Adj count 0
Hello 10, Dead 40, ReXmit 5, Not Stub
Auth type: MD5, Active key ID 3, Start time 2002 Nov 19 10:00:00 PST
IPsec SA Name: sa

show ospf3 interface detail

```
user@host> show ospf3 interface so-0/0/3.0 detail
```

| Interface | State | Area | DR-ID | BDR-ID | Nbrs |
|------------|--------|---------|---------|---------|------|
| so-0/0/3.0 | PtToPt | 0.0.0.0 | 0.0.0.0 | 0.0.0.0 | 1 |

Address fe80::2a0:a5ff:fe28:1dfc, Prefix-length 64
OSPF3-Intf-index 1, Type P2P, MTU 4470, Cost 12, Adj-count 1
Hello 10, Dead 40, ReXmit 5, Not Stub

show ospf interface detail (When Multiarea Adjacency Is Configured)

```
user@host> show ospf interface detail
```

```
regress@router> show ospf interface detail
```

| Interface | State | Area | DR ID | BDR ID | Nbrs |
|-----------|-------|---------|--------------|---------|------|
| lo0.0 | DR | 0.0.0.0 | 10.255.245.2 | 0.0.0.0 | 0 |

Type: LAN, Address: 127.0.0.1, Mask: 255.255.255.255, MTU: 65535, Cost: 0
DR addr: 127.0.0.1, Adj count: 0, Priority: 128
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Cost: 0

| Interface | State | Area | DR ID | BDR ID | Nbrs |
|-----------|-------|---------|--------------|---------|------|
| lo0.0 | DR | 0.0.0.0 | 10.255.245.2 | 0.0.0.0 | 0 |

Type: LAN, Address: 10.255.245.2, Mask: 255.255.255.255, MTU: 65535, Cost: 0
DR addr: 10.255.245.2, Adj count: 0, Priority: 128
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Cost: 0

| Interface | State | Area | DR ID | BDR ID | Nbrs |
|------------|--------|---------|---------|---------|------|
| so-0/0/0.0 | PtToPt | 0.0.0.0 | 0.0.0.0 | 0.0.0.0 | 1 |

Type: P2P, Address: 0.0.0.0, Mask: 0.0.0.0, MTU: 4470, Cost: 1
Adj count: 1
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Cost: 1

| Interface | State | Area | DR ID | BDR ID | Nbrs |
|------------|--------|---------|---------|---------|------|
| so-0/0/0.0 | PtToPt | 0.0.0.0 | 0.0.0.0 | 0.0.0.0 | 0 |

```

Type: P2P, Address: 192.168.37.46, Mask: 255.255.255.254, MTU: 4470, Cost: 1
Adj count: 0, , Passive
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Passive, Cost: 1
so-1/0/0.0      PtToPt  0.0.0.0      0.0.0.0      0.0.0.0      1

Type: P2P, Address: 0.0.0.0, Mask: 0.0.0.0, MTU: 4470, Cost: 1
Adj count: 1
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Cost: 1
so-1/0/0.0      PtToPt  0.0.0.0      0.0.0.0      0.0.0.0      0

Type: P2P, Address: 192.168.37.54, Mask: 255.255.255.254, MTU: 4470, Cost: 1
Adj count: 0, , Passive
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Passive, Cost: 1
so-0/0/0.0      PtToPt  1.1.1.1      0.0.0.0      0.0.0.0      1

Type: P2P, Address: 0.0.0.0, Mask: 0.0.0.0, MTU: 4470, Cost: 1
Adj count: 1, Secondary
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Cost: 1
so-1/0/0.0      PtToPt  1.1.1.1      0.0.0.0      0.0.0.0      1

Type: P2P, Address: 0.0.0.0, Mask: 0.0.0.0, MTU: 4470, Cost: 1
Adj count: 1, Secondary
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Cost: 1
so-0/0/0.0      PtToPt  2.2.2.2      0.0.0.0      0.0.0.0      1

Type: P2P, Address: 0.0.0.0, Mask: 0.0.0.0, MTU: 4470, Cost: 1
Adj count: 1, Secondary
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Cost: 1
so-1/0/0.0      PtToPt  2.2.2.2      0.0.0.0      0.0.0.0      1

Type: P2P, Address: 0.0.0.0, Mask: 0.0.0.0, MTU: 4470, Cost: 1
Adj count: 1, Secondary
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Cost: 1

```

show ospf interface area area-id

```

user@host> show ospf interface area 1.1.1.1
Interface      State  Area      DR ID      BDR ID      Nbrs
so-0/0/0.0     PtToPt 1.1.1.1   0.0.0.0    0.0.0.0     1
so-1/0/0.0     PtToPt 1.1.1.1   0.0.0.0    0.0.0.0     1

```

show ospf interface extensive (When Flooding Reduction Is Enabled)

```

user@host> show ospf interface extensive
Interface      State  Area      DR ID      BDR ID      Nbrs
fe-0/0/0.0     PtToPt 0.0.0.0    0.0.0.0    0.0.0.0     0

Type: P2P, Address: 10.10.10.1, Mask: 255.255.255.0, MTU: 1500, Cost: 1
Adj count: 0

```

Secondary, Flood Reduction
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None
Topology default (ID 0) -> Cost: 1

show ospf interface
extensive
(When LDP
Synchronization Is
Configured)

```
user@host> show ospf interface extensive
Interface          State      Area          DR ID          BDR ID
Nbrs
so-1/0/3.0         Down      0.0.0.0       0.0.0.0       0.0.0.0
0
  Type: P2P, Address: 0.0.0.0, Mask: 0.0.0.0, MTU: 4470, Cost: 65535
  Adj count: 0
  Hello: 10, Dead: 40, ReXmit: 5, Not Stub
  Auth type: None
  LDP sync state: in holddown, for: 00:00:08, reason: LDP down during config
                    config holddtime: 10 seconds, remaining: 1
```

show (ospf | ospf3) neighbor

| | |
|---|---|
| Syntax | <pre>show (ospf ospf3) neighbor <brief detail extensive> <area <i>area-id</i>> <instance (all <i>instance-name</i>)> <interface <i>interface-name</i>> <logical-system (all <i>logical-system-name</i>)> <neighbor> <realm (ipv4-multicast ipv4-unicast ipv6-multicast)></pre> |
| Syntax (EX Series Switches and QFX Series) | <pre>show (ospf ospf3) neighbor <brief detail extensive> <area <i>area-id</i>> <instance (all <i>instance-name</i>)> <interface <i>interface-name</i>> <neighbor></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>instance all option introduced in Junos OS Release 9.1.</p> <p>instance all option introduced in Junos OS Release 9.1 for EX Series switches.</p> <p>area, interface, and realm options introduced in Junos OS Release 9.2.</p> <p>area and interface options introduced in Junos OS Release 9.2 for EX Series switches.</p> <p>Command introduced in Junos OS Release 11.3 for the QFX Series.</p> |
| Description | <p>Display information about OSPF neighbors.</p> <p>CPU utilization might increase while the device learns its OSPF neighbors. We recommend that you use the show (ospf ospf3) neighbor command after the device learns and establishes OSPF neighbor adjacencies. Depending on the size of your network, this might take several minutes. If you receive a “timeout communicating with routing daemon” error when using the show (ospf ospf3) neighbor command, wait several minutes before attempting to use the command again. This is not a critical system error, but you might experience a delay in using the CLI.</p> |
| Options | <p>none—Display standard information about all OSPF neighbors for all routing instances.</p> <p>brief detail extensive—(Optional) Display the specified level of output.</p> <p>area <i>area-id</i>—(Optional) Display information about the OSPF neighbors for the specified area.</p> <p>instance (all <i>instance-name</i>)—(Optional) Display all OSPF interfaces for all routing instances or under the named routing instance.</p> <p>interface <i>interface-name</i>—(Optional) Display information about OSPF neighbors for the specified logical interface.</p> <p>logical-system (all <i>logical-system-name</i>)—(Optional) Perform this operation on all logical systems or on a particular logical system.</p> |

neighbor—(Optional) Display information about the specified OSPF neighbor.

realm (ipv4-multicast | ipv4-unicast | ipv6-multicast)—(OSPFv3 only) (Optional) Display information about the OSPF neighbors for the specified OSPFv3 realm, or address family. Use the **realm** option to specify an address family for OSPFv3 other than IPv6 unicast, which is the default.

Required Privilege Level view

Related Documentation

- [clear \(ospf | ospf3\) neighbor](#)

List of Sample Output

- [show ospf neighbor brief on page 110](#)
- [show ospf neighbor detail on page 110](#)
- [show ospf neighbor extensive on page 110](#)
- [show ospf3 neighbor detail on page 111](#)
- [show ospf neighbor area area-id on page 111](#)
- [show ospf neighbor interface interface-name on page 111](#)
- [show ospf3 neighbor instance all \(OSPFv3 Multiple Family Address Support Enabled\) on page 112](#)

Output Fields [Table 7 on page 107](#) lists the output fields for the **show (ospf | ospf3) neighbor** command. Output fields are listed in the approximate order in which they appear.

Table 7: show (ospf | ospf3) neighbor Output Fields

| Field Name | Field Description | Level of Output |
|------------------|--|-----------------|
| Address | Address of the neighbor. | All levels |
| Interface | Interface through which the neighbor is reachable. | All levels |

Table 7: show (ospf | ospf3) neighbor Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|---------------------------------------|--|-------------------------|
| State | <p>State of the neighbor:</p> <ul style="list-style-type: none"> • Attempt—Valid only for neighbors attached to nonbroadcast networks. It indicates that no recent information has been received from the neighbor, but that a more concerted effort must be made to contact the neighbor. • Down—Initial state of a neighbor conversation. It indicates that no recent information has been received from the neighbor. Hello packets might continue to be sent to neighbors in the Down state, although at a reduced frequency. • Exchange—Routing device is describing its entire link-state database by sending database description packets to the neighbor. Each packet has a sequence number and is explicitly acknowledged. • ExStart—First step in creating an adjacency between the two neighboring routing devices. The goal of this step is to determine which routing device is the master, and to determine the initial sequence number. • Full—Neighboring routing devices are fully adjacent. These adjacencies appear in router link and network link advertisements. • Init—A hello packet has recently been sent by the neighbor. However, bidirectional communication has not yet been established with the neighbor. This state might occur, for example, because the routing device itself did not appear in the neighbor's hello packet. • Loading—Link-state request packets are sent to the neighbor to acquire more recent advertisements that have been discovered (but not yet received) in the Exchange state. • 2Way—Communication between the two routing devices is bidirectional. This state has been ensured by the operation of the Hello Protocol. This is the most advanced state short of beginning adjacency establishment. The (backup) designated router is selected from the set of neighbors in state 2Way or greater. | All levels |
| ID | Router ID of the neighbor. | All levels |
| Pri | Priority of the neighbor to become the designated router. | All levels |
| Dead | Number of seconds until the neighbor becomes unreachable. | All levels |
| Link state acknowledgment list | Number of link-state acknowledgments received. | extensive |
| Link state retransmission list | <p>Total number of link-state advertisements retransmitted. For extensive output only, the following information is also displayed:</p> <ul style="list-style-type: none"> • Type—Type of link advertisement: ASBR, Sum, Extern, Network, NSSA, OpaqueArea, Router, or Summary. • LSA ID—LSA identifier included in the advertisement. An asterisk preceding the identifier marks database entries that originated from the local routing device. • Adv rtr—Address of the routing device that sent the advertisement. • Seq—Link sequence number of the advertisement. | detail extensive |

Table 7: show (ospf | ospf3) neighbor Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|-------------------------|--|-------------------------|
| Neighbor-address | (OSPFv3 only) If the neighbor uses virtual links, the Neighbor-address is the site-local, local, or global address. If the neighbor uses a physical interface, the Neighbor-address is an IPv6 link-local address. | detail extensive |
| area | Area that the neighbor is in. | detail extensive |
| OSPF3-Intf-Index | (OSPFv3 only) Displays the OSPFv3 interface index. | detail extensive |
| opt | Option bits received in the hello packets from the neighbor. | detail extensive |
| DR or DR-ID | Address of the designated router. | detail extensive |
| BDR or BDR-ID | Address of the backup designated router. | detail extensive |
| Up | Length of time since the neighbor came up. | detail extensive |
| adjacent | Length of time since the adjacency with the neighbor was established. | detail extensive |

Sample Output

show ospf neighbor brief

```
user@host> show ospf neighbor brief
```

| Address | Intf | State | ID | Pri | Dead |
|-----------------|--------|-------|---------------|-----|------|
| 192.168.254.225 | fxp3.0 | 2Way | 10.250.240.32 | 128 | 36 |
| 192.168.254.230 | fxp3.0 | Full | 10.250.240.8 | 128 | 38 |
| 192.168.254.229 | fxp3.0 | Full | 10.250.240.35 | 128 | 33 |
| 10.1.1.129 | fxp2.0 | Full | 10.250.240.12 | 128 | 37 |
| 10.1.1.131 | fxp2.0 | Full | 10.250.240.11 | 128 | 38 |
| 10.1.2.1 | fxp1.0 | Full | 10.250.240.9 | 128 | 32 |
| 10.1.2.81 | fxp0.0 | Full | 10.250.240.10 | 128 | 33 |

show ospf neighbor detail

```
user@host> show ospf neighbor detail
```

| Address | Interface | State | ID | Pri | Dead |
|---|-------------|---------|-----------|-----|------|
| 10.5.1.2 | ge-1/2/0.1 | Full | 10.5.1.2 | 128 | 37 |
| area 0.0.0.1, opt 0x42, DR 10.5.1.2, BDR 10.5.1.1 | | | | | |
| Up 06:09:28, adjacent 05:17:36 | | | | | |
| Link state acknowledgment list: 3 entries | | | | | |
| Link state retransmission list: 9 entries | | | | | |
| 10.5.10.2 | ge-1/2/0.10 | ExStart | 10.5.1.38 | 128 | 34 |
| area 0.0.0.1, opt 0x42, DR 10.5.10.2, BDR 10.5.10.1 | | | | | |
| Up 06:09:28 | | | | | |
| master, seq 0xac1530f8, rexmit DBD in 3 sec | | | | | |
| rexmit LSREQ in 0 sec | | | | | |
| 10.5.11.2 | ge-1/2/0.11 | Full | 10.5.1.42 | 128 | 38 |
| area 0.0.0.1, opt 0x42, DR 10.5.11.2, BDR 10.5.11.1 | | | | | |
| Up 06:09:28, adjacent 05:26:46 | | | | | |
| Link state retransmission list: 1 entries | | | | | |
| 10.5.12.2 | ge-1/2/0.12 | ExStart | 10.5.1.46 | 128 | 33 |
| area 0.0.0.1, opt 0x42, DR 10.5.12.2, BDR 10.5.12.1 | | | | | |
| Up 06:09:28 | | | | | |
| master, seq 0xac188a68, rexmit DBD in 2 sec | | | | | |
| rexmit LSREQ in 0 sec | | | | | |

show ospf neighbor extensive

```
user@host> show ospf neighbor extensive
```

| Address | Interface | State | ID | Pri | Dead |
|---|------------|--------------|------------|-----|------|
| 10.5.1.2 | ge-1/2/0.1 | Full | 10.5.1.2 | 128 | 33 |
| area 0.0.0.1, opt 0x42, DR 10.5.1.2, BDR 10.5.1.1 | | | | | |
| Up 06:09:42, adjacent 05:17:50 | | | | | |
| Link state retransmission list: | | | | | |
| Type | LSA ID | Adv rtr | Seq | | |
| Summary | 10.8.56.0 | 172.25.27.82 | 0x8000004d | | |
| Router | 10.5.1.94 | 10.5.1.94 | 0x8000005c | | |
| Network | 10.5.24.2 | 10.5.1.94 | 0x80000036 | | |
| Summary | 10.8.57.0 | 172.25.27.82 | 0x80000024 | | |
| Extern | 1.10.90.0 | 10.8.1.2 | 0x80000041 | | |
| Extern | 1.4.109.0 | 10.6.1.2 | 0x80000041 | | |

```

Router 10.5.1.190      10.5.1.190      0x8000005f
Network 10.5.48.2      10.5.1.190      0x8000003d
Summary 10.8.58.0      172.25.27.82    0x8000004d
Extern 1.10.91.0      10.8.1.2        0x80000041
Extern 1.4.110.0      10.6.1.2        0x80000041
Router 10.5.1.18      10.5.1.18      0x8000005f
Network 10.5.5.2      10.5.1.18      0x80000033
Summary 10.8.59.0      172.25.27.82    0x8000003a
Summary 10.8.62.0      172.25.27.82    0x80000025

10.5.10.2      ge-1/2/0.10      ExStart 10.5.1.38      128 38
area 0.0.0.1, opt 0x42, DR 10.5.10.2, BDR 10.5.10.1
Up 06:09:42
master, seq 0xac1530f8, retransmit DBD in 2 sec
retransmit LSREQ in 0 sec
10.5.11.2      ge-1/2/0.11      Full 10.5.1.42      128 33
area 0.0.0.1, opt 0x42, DR 10.5.11.2, BDR 10.5.11.1
Up 06:09:42, adjacent 05:27:00
Link state retransmission list:

Type      LSA ID      Adv rtr      Seq
Summary 10.8.58.0      172.25.27.82    0x8000004d
Extern 1.10.91.0      10.8.1.2        0x80000041
Extern 1.1.247.0      10.5.1.2        0x8000003f
Extern 1.4.110.0      10.6.1.2        0x80000041
Router 10.5.1.18      10.5.1.18      0x8000005f
Network 10.5.5.2      10.5.1.18      0x80000033
Summary 10.8.59.0      172.25.27.82    0x8000003a

```

show ospf3 neighbor detail

```

user@host> show ospf3 neighbor detail
ID      Interface      State      Pri  Dead
10.255.71.13  fe-0/0/2.0      Full      128  30
Neighbor-address fe80::290:69ff:fe9b:e002
area 0.0.0.0, opt 0x13, OSPF3-Intf-Index 2
DR-ID 10.255.71.13, BDR-ID 10.255.71.12
Up 02:51:43, adjacent 02:51:43

```

show ospf neighbor area area-id

```

user@host >show ospf neighbor area 1.1.1.1
Address      Interface      State      ID      Pri  Dead
192.168.37.47  so-0/0/0.0      Full      10.255.245.4  128  33
Area 1.1.1.1
192.168.37.55  so-1/0/0.0      Full      10.255.245.5  128  37
Area 1.1.1.1

```

**show ospf neighbor
interface
interface-name**

```
user@host > show ospf neighbor interface so-0/0/0.0
Address      Interface      State      ID            Pri    Dead
192.168.37.47 so-0/0/0.0     Full      10.255.245.4  128    37
  Area 0.0.0.0
192.168.37.47 so-0/0/0.0     Full      10.255.245.4  128    33
  Area 1.1.1.1
192.168.37.47 so-0/0/0.0     Full      10.255.245.4  128    32
  Area 2.2.2.2
```

**show ospf3 neighbor
instance all (OSPFv3
Multiple Family
Address Support
Enabled)**

```
user @host > show ospf3 neighbor instance all
Instance: ina
  Realm: ipv6-unicast
    ID      Interface      State      Pri    Dead
    100.1.1.1 fe-0/0/2.0     Full      128    37
    Neighbor-address fe80::217:cb00:c87c:8c03
Instance: inb
  Realm: ipv4-unicast
    ID      Interface      State      Pri    Dead
    100.1.2.1 fe-0/0/2.1     Full      128    33
    Neighbor-address fe80::217:cb00:c97c:8c03
```

show route

| | |
|------------------------------------|--|
| Syntax | <pre>show route <all> <destination-prefix> <logical-system (all logical-system-name)> <private></pre> |
| Syntax (EX Series Switches) | <pre>show route <all> <destination-prefix> <private></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>Option private introduced in Junos OS Release 9.5.</p> <p>Option private introduced in Junos OS Release 9.5 for EX Series switches.</p> |
| Description | Display the active entries in the routing tables. |
| Options | <p>none—Display brief information about all active entries in the routing tables.</p> <p>all—(Optional) Display information about all routing tables, including private, or internal, routing tables.</p> <p>destination-prefix—(Optional) Display active entries for the specified address or range of addresses.</p> <p>logical-system (all logical-system-name)—(Optional) Perform this operation on all logical systems or on a particular logical system.</p> <p>private—(Optional) Display information only about all private, or internal, routing tables.</p> |
| Required Privilege Level | view |
| Related Documentation | <ul style="list-style-type: none"> • Example: Configuring RIP • Example: Configuring RIPng • Example: Configuring IS-IS • Examples: Configuring Internal BGP Peering • Examples: Configuring External BGP Peering • Examples: Configuring OSPF Routing Policy |
| List of Sample Output | <p>show route on page 117</p> <p>show route destination-prefix on page 117</p> <p>show route extensive on page 117</p> |

Output Fields Table 8 on page 114 describes the output fields for the **show route** command. Output fields are listed in the approximate order in which they appear.

Table 8: show route Output Fields

| Field Name | Field Description |
|---------------------------------|--|
| <i>routing-table-name</i> | Name of the routing table (for example, inet.0). |
| <i>number destinations</i> | Number of destinations for which there are routes in the routing table. |
| <i>number routes</i> | <p>Number of routes in the routing table and total number of routes in the following states:</p> <ul style="list-style-type: none"> • active (routes that are active). • holddown (routes that are in the pending state before being declared inactive). A holddown route was once the active route and is no longer the active route. The route is in the holddown state because a protocol still has interest in the route, meaning that the interest bit is set. A protocol might have its interest bit set on the previously active route because the protocol is still advertising the route. The route will be deleted after all protocols withdraw their advertisement of the route and remove their interest bit. A persistent holddown state often means that the interested protocol is not releasing its interest bit properly. <p>However, if you have configured advertisement of multiple routes (with the add-path or advertise-inactive statement), the holddown bit is most likely set because BGP is advertising the route as an active route. In this case, you can ignore the holddown state because nothing is wrong.</p> <ul style="list-style-type: none"> • hidden (routes that are not used because of a routing policy). |
| <i>destination-prefix</i> | <p>Route destination (for example:10.0.0.1/24). Sometimes the route information is presented in another format, such as:</p> <ul style="list-style-type: none"> • MPLS-label (for example, 80001). • interface-name (for example, ge-1/0/2). • neighbor-address:control-word-status:encapsulation type:vc-id:source (Layer 2 circuit only. For example, 10.1.1.195:NoCtrlWord:1:1:Local/96): <ul style="list-style-type: none"> • neighbor-address—Address of the neighbor. • control-word-status—Whether the use of the control word has been negotiated for this virtual circuit: NoCtrlWord or CtrlWord. • encapsulation type—Type of encapsulation, represented by a number: (1) Frame Relay DLCI, (2) ATM AAL5 VCC transport, (3) ATM transparent cell transport, (4) Ethernet, (5) VLAN Ethernet, (6) HDLC, (7) PPP, (8) ATM VCC cell transport, (10) ATM VPC cell transport. • vc-id—Virtual circuit identifier. • source—Source of the advertisement: Local or Remote. |
| [protocol, preference] | <p>Protocol from which the route was learned and the preference value for the route.</p> <ul style="list-style-type: none"> • +—A plus sign indicates the active route, which is the route installed from the routing table into the forwarding table. • -—A hyphen indicates the last active route. • *—An asterisk indicates that the route is both the active and the last active route. An asterisk before a to line indicates the best subpath to the route. <p>In every routing metric except for the BGP LocalPref attribute, a lesser value is preferred. In order to use common comparison routines, Junos OS stores the 1's complement of the LocalPref value in the Preference2 field. For example, if the LocalPref value for Route 1 is 100, the Preference2 value is -101. If the LocalPref value for Route 2 is 155, the Preference2 value is -156. Route 2 is preferred because it has a higher LocalPref value and a lower Preference2 value.</p> |

Table 8: show route Output Fields (*continued*)

| Field Name | Field Description |
|---|---|
| <i>weeks:days</i> <i>hours:minutes:seconds</i> | How long the route been known (for example, 2w4d 13:11:14 , or 2 weeks, 4 days, 13 hours, 11 minutes, and 14 seconds). |
| metric | Cost value of the indicated route. For routes within an AS, the cost is determined by the IGP and the individual protocol metrics. For external routes, destinations, or routing domains, the cost is determined by a preference value. |
| localpref | Local preference value included in the route. |
| from | Interface from which the route was received. |
| AS path | <p>AS path through which the route was learned. The letters at the end of the AS path indicate the path origin, providing an indication of the state of the route at the point at which the AS path originated:</p> <ul style="list-style-type: none"> • I—IGP. • E—EGP. • ?—Incomplete; typically, the AS path was aggregated. <p>When AS path numbers are included in the route, the format is as follows:</p> <ul style="list-style-type: none"> • []—Brackets enclose the local AS number associated with the AS path if more than one AS number is configured on the routing device, or if AS path prepending is configured. • { }—Braces enclose AS sets, which are groups of AS numbers in which the order does not matter. A set commonly results from route aggregation. The numbers in each AS set are displayed in ascending order. • ()—Parentheses enclose a confederation. • ([])—Parentheses and brackets enclose a confederation set. <p>NOTE: In Junos OS Release 10.3 and later, the AS path field displays an unrecognized attribute and associated hexadecimal value if BGP receives attribute 128 (attribute set) and you have not configured an independent domain in any routing instance.</p> |
| validation-state | <p>(BGP-learned routes) Validation status of the route:</p> <ul style="list-style-type: none"> • Invalid—Indicates that the prefix is found, but either the corresponding AS received from the EBGP peer is not the AS that appears in the database, or the prefix length in the BGP update message is longer than the maximum length permitted in the database. • Unknown—Indicates that the prefix is not among the prefixes or prefix ranges in the database. • Valid—Indicates that the prefix and autonomous system pair are found in the database. |
| to | <p>Next hop to the destination. An angle bracket (>) indicates that the route is the selected route.</p> <p>If the destination is Discard, traffic is dropped.</p> |

Table 8: show route Output Fields (*continued*)

| Field Name | Field Description |
|------------|---|
| via | <p>Interface used to reach the next hop. If there is more than one interface available to the next hop, the interface that is actually used is followed by the word Selected. This field can also contain the following information:</p> <ul style="list-style-type: none">• Weight—Value used to distinguish primary, secondary, and fast reroute backup routes. Weight information is available when MPLS label-switched path (LSP) link protection, node-link protection, or fast reroute is enabled, or when the standby state is enabled for secondary paths. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible.• Balance—Balance coefficient indicating how traffic of unequal cost is distributed among next hops when a routing device is performing unequal-cost load balancing. This information is available when you enable BGP multipath load balancing.• lsp-path-name—Name of the LSP used to reach the next hop.• label-action—MPLS label and operation occurring at the next hop. The operation can be pop (where a label is removed from the top of the stack), push (where another label is added to the label stack), or swap (where a label is replaced by another label). |

Sample Output

show route

```
user@host> show route
inet.0: 11 destinations, 12 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1:65500:1:10.0.0.20/240
    *[MVPN/70] 19:53:41, metric2 1
    Indirect
1:65500:1:10.0.0.40/240
    *[BGP/170] 19:53:29, localpref 100, from 10.0.0.30
    AS path: I
    > to 10.0.24.4 via lt-0/3/0.24, label-switched-path toD
    [BGP/170] 19:53:26, localpref 100, from 10.0.0.33
    AS path: I
    > to 10.0.24.4 via lt-0/3/0.24, label-switched-path toD
1:65500:1:10.0.0.60/240
    *[BGP/170] 19:53:29, localpref 100, from 10.0.0.30
    AS path: I
    > to 10.0.28.8 via lt-0/3/0.28, label-switched-path toF
    [BGP/170] 19:53:25, localpref 100, from 10.0.0.33
    AS path: I
    > to 10.0.28.8 via lt-0/3/0.28, label-switched-path toF
```

show route destination-prefix

```
user@host> show route 172.16.0.0/12

inet.0: 10 destinations, 10 routes (9 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.0.0/12    *[Static/5] 2w4d 12:54:27
    > to 192.168.167.254 via fxp0.0
```

show route extensive

```
user@host> show route extensive
v1.mvpn.0: 5 destinations, 8 routes (5 active, 1 holddown, 0 hidden)
1:65500:1:10.0.0.40/240 (1 entry, 1 announced)
    *BGP Preference: 170/-101
    PMSI: Flags 0x0: Label[0:0:0]: PIM-SM: Sender 10.0.0.40 Group 225.1.1.1

    Next hop type: Indirect
    Address: 0x92455b8
    Next-hop reference count: 2
    Source: 10.0.0.30
    Protocol next hop: 10.0.0.40
    Indirect next hop: 2 no-forward
    State: <Active Int Ext>
        Local AS: 65500 Peer AS: 65500
    Age: 3 Metric2: 1
    Task: BGP_65500.10.0.0.30+179
    Announcement bits (2): 0-PIM.v1 1-mvpn global task
    AS path: I (Originator) Cluster list: 10.0.0.30
    AS path: Originator ID: 10.0.0.40
    Communities: target:65520:100
    Import Accepted
    Localpref: 100
    Router ID: 10.0.0.30
    Primary Routing Table bgp.mvpn.0
    Indirect next hops: 1
        Protocol next hop: 10.0.0.40 Metric: 1
```

```
Indirect next hop: 2 no-forward
Indirect path forwarding next hops: 1
  Next hop type: Router
  Next hop: 10.0.24.4 via lt-0/3/0.24 weight 0x1
10.0.0.40/32 Originating RIB: inet.3
  Metric: 1                               Node path count: 1
  Forwarding nexthops: 1
    Nexthop: 10.0.24.4 via lt-0/3/0.24
```

show route detail

| | |
|------------------------------------|---|
| Syntax | show route detail <destination-prefix> <logical-system (all logical-system-name)> |
| Syntax (EX Series Switches) | show route detail <destination-prefix> |
| Release Information | Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.0 for EX Series switches. |
| Description | Display detailed information about the active entries in the routing tables. |
| Options | <p>none—Display all active entries in the routing table on all systems.</p> <p>destination-prefix—(Optional) Display active entries for the specified address or range of addresses.</p> <p>logical-system (all logical-system-name)—(Optional) Perform this operation on all logical systems or on a particular logical system.</p> |
| Required Privilege Level | view |
| List of Sample Output | show route detail on page 128 show route detail (with BGP Multipath) on page 133 |
| Output Fields | Table 9 on page 119 describes the output fields for the show route detail command. Output fields are listed in the approximate order in which they appear. |

Table 9: show route detail Output Fields

| Field Name | Field Description |
|----------------------------|---|
| <i>routing-table-name</i> | Name of the routing table (for example, inet.0). |
| <i>number destinations</i> | Number of destinations for which there are routes in the routing table. |
| <i>number routes</i> | Number of routes in the routing table and total number of routes in the following states: <ul style="list-style-type: none"> active (routes that are active) holddown (routes that are in the pending state before being declared inactive) hidden (routes that are not used because of a routing policy) |

Table 9: show route detail Output Fields (*continued*)

| Field Name | Field Description |
|--|---|
| <i>route-destination</i> (entry, announced) | <p>Route destination (for example:10.0.0.1/24). The entry value is the number of routes for this destination, and the announced value is the number of routes being announced for this destination. Sometimes the route destination is presented in another format, such as:</p> <ul style="list-style-type: none"> • MPLS-label (for example, 80001). • interface-name (for example, ge-1/0/2). • neighbor-address:control-word-status:encapsulation type:vc-id:source (Layer 2 circuit only; for example, 10.1.1.195:NoCtrlWord:1:1:Local/96). <ul style="list-style-type: none"> • neighbor-address—Address of the neighbor. • control-word-status—Whether the use of the control word has been negotiated for this virtual circuit: NoCtrlWord or CtrlWord. • encapsulation type—Type of encapsulation, represented by a number: (1) Frame Relay DLCI, (2) ATM AAL5 VCC transport, (3) ATM transparent cell transport, (4) Ethernet, (5) VLAN Ethernet, (6) HDLC, (7) PPP, (8) ATM VCC cell transport, (10) ATM VPC cell transport. • vc-id—Virtual circuit identifier. • source—Source of the advertisement: Local or Remote. |
| label stacking | <p>(Next-to-the-last-hop routing device for MPLS only) Depth of the MPLS label stack, where the label-popping operation is needed to remove one or more labels from the top of the stack. A pair of routes is displayed, because the pop operation is performed only when the stack depth is two or more labels.</p> <ul style="list-style-type: none"> • S=0 route indicates that a packet with an incoming label stack depth of 2 or more exits this routing device with one fewer label (the label-popping operation is performed). • If there is no S= information, the route is a normal MPLS route, which has a stack depth of 1 (the label-popping operation is not performed). |
| [<i>protocol, preference</i>] | <p>Protocol from which the route was learned and the preference value for the route.</p> <ul style="list-style-type: none"> • +—A plus sign indicates the active route, which is the route installed from the routing table into the forwarding table. • - —A hyphen indicates the last active route. • *—An asterisk indicates that the route is both the active and the last active route. An asterisk before a to line indicates the best subpath to the route. <p>In every routing metric except for the BGP LocalPref attribute, a lesser value is preferred. In order to use common comparison routines, Junos OS stores the 1's complement of the LocalPref value in the Preference2 field. For example, if the LocalPref value for Route 1 is 100, the Preference2 value is -101. If the LocalPref value for Route 2 is 155, the Preference2 value is -156. Route 2 is preferred because it has a higher LocalPref value and a lower Preference2 value.</p> |
| Level | <p>(IS-IS only). 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.</p> |
| Route Distinguisher | IP subnet augmented with a 64-bit prefix. |
| Next-hop type | Type of next hop. For a description of possible values for this field, see Table 10 on page 123 . |

Table 9: show route detail Output Fields (*continued*)

| Field Name | Field Description |
|--|---|
| Next-hop reference count | Number of references made to the next hop. |
| Flood nexthop branches exceed maximum message | Indicates that the number of flood next-hop branches exceeded the system limit of 32 branches, and only a subset of the flood next-hop branches were installed in the kernel. |
| Source | IP address of the route source. |
| Next hop | Network layer address of the directly reachable neighboring system. |
| via | <p>Interface used to reach the next hop. If there is more than one interface available to the next hop, the name of the interface that is actually used is followed by the word Selected. This field can also contain the following information:</p> <ul style="list-style-type: none"> • Weight—Value used to distinguish primary, secondary, and fast reroute backup routes. Weight information is available when MPLS label-switched path (LSP) link protection, node-link protection, or fast reroute is enabled, or when the standby state is enabled for secondary paths. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible. • Balance—Balance coefficient indicating how traffic of unequal cost is distributed among next hops when a routing device is performing unequal-cost load balancing. This information is available when you enable BGP multipath load balancing. |
| Label-switched-path lsp-path-name | Name of the LSP used to reach the next hop. |
| Label operation | MPLS label and operation occurring at this routing device. The operation can be pop (where a label is removed from the top of the stack), push (where another label is added to the label stack), or swap (where a label is replaced by another label). |
| Interface | (Local only) Local interface name. |
| Protocol next hop | Network layer address of the remote routing device that advertised the prefix. This address is used to derive a forwarding next hop. |
| Indirect next hop | Index designation used to specify the mapping between protocol next hops, tags, kernel export policy, and the forwarding next hops. |
| State | State of the route (a route can be in more than one state). See Table 11 on page 125 . |
| Local AS | AS number of the local routing device. |
| Age | How long the route has been known. |
| AIGP | Accumulated interior gateway protocol (AIGP) BGP attribute. |
| Metricn | Cost value of the indicated route. For routes within an AS, the cost is determined by IGP and the individual protocol metrics. For external routes, destinations, or routing domains, the cost is determined by a preference value. |

Table 9: show route detail Output Fields (*continued*)

| Field Name | Field Description |
|--------------------------------|---|
| MED-plus-IGP | Metric value for BGP path selection to which the IGP cost to the next-hop destination has been added. |
| TTL-Action | <p>For MPLS LSPs, state of the TTL propagation attribute. Can be enabled or disabled for all RSVP-signaled and LDP-signaled LSPs or for specific VRF routing instances.</p> <p>For sample output, see show route table.</p> |
| Task | Name of the protocol that has added the route. |
| Announcement bits | List of protocols that announce this route. n-Resolve inet indicates that the route is used for route resolution for next hops found in the routing table. n is an index used by Juniper Networks customer support only. |
| AS path | <p>AS path through which the route was learned. The letters at the end of the AS path indicate the path origin, providing an indication of the state of the route at the point at which the AS path originated:</p> <ul style="list-style-type: none"> • I—IGP. • E—EGP. • Recorded—The AS path is recorded by the sample process (sampled). • ?—Incomplete; typically, the AS path was aggregated. <p>When AS path numbers are included in the route, the format is as follows:</p> <ul style="list-style-type: none"> • []—Brackets enclose the number that precedes the AS path. This number represents the number of ASs present in the AS path, when calculated as defined in RFC 4271. This value is used in the AS-path merge process, as defined in RFC 4893. • []—If more than one AS number is configured on the routing device, or if AS path prepending is configured, brackets enclose the local AS number associated with the AS path. • { }—Braces enclose AS sets, which are groups of AS numbers in which the order does not matter. A set commonly results from route aggregation. The numbers in each AS set are displayed in ascending order. • ()—Parentheses enclose a confederation. • ([])—Parentheses and brackets enclose a confederation set. <p>NOTE: In Junos OS Release 10.3 and later, the AS path field displays an unrecognized attribute and associated hexadecimal value if BGP receives attribute 128 (attribute set) and you have not configured an independent domain in any routing instance.</p> |
| VC Label | MPLS label assigned to the Layer 2 circuit virtual connection. |
| MTU | Maximum transmission unit (MTU) of the Layer 2 circuit. |
| VLAN ID | VLAN identifier of the Layer 2 circuit. |
| Prefixes bound to route | Forwarding equivalent class (FEC) bound to this route. Applicable only to routes installed by LDP. |
| Communities | Community path attribute for the route. See Table 12 on page 127 for all possible values for this field. |
| Layer2-info: encaps | Layer 2 encapsulation (for example, VPLS). |

Table 9: show route detail Output Fields (*continued*)

| Field Name | Field Description |
|----------------------------------|--|
| control flags | Control flags: none or Site Down . |
| mtu | Maximum transmission unit (MTU) information. |
| Label-Base, range | First label in a block of labels and label block size. A remote PE routing device uses this first label when sending traffic toward the advertising PE routing device. |
| status vector | Layer 2 VPN and VPLS network layer reachability information (NLRI). |
| Accepted Multipath | Current active path when BGP multipath is configured. |
| Accepted MultipathContrib | Path currently contributing to BGP multipath. |
| Localpref | Local preference value included in the route. |
| Router ID | BGP router ID as advertised by the neighbor in the open message. |
| Primary Routing Table | In a routing table group, the name of the primary routing table in which the route resides. |
| Secondary Tables | In a routing table group, the name of one or more secondary tables in which the route resides. |

Table 10 on page 123 describes all possible values for the **Next-hop Types** output field.

Table 10: Next-hop Types Output Field Values

| Next-Hop Type | Description |
|--------------------------|--|
| Broadcast (bcast) | Broadcast next hop. |
| Deny | Deny next hop. |
| Discard | Discard next hop. |
| Flood | Flood next hop. Consists of components called branches, up to a maximum of 32 branches. Each flood next-hop branch sends a copy of the traffic to the forwarding interface. Used by point-to-multipoint RSVP, point-to-multipoint LDP, point-to-multipoint CCC, and multicast. |
| Hold | Next hop is waiting to be resolved into a unicast or multicast type. |
| Indexed (idxd) | Indexed next hop. |

Table 10: Next-hop Types Output Field Values (*continued*)

| Next-Hop Type | Description |
|---------------------------------|--|
| Indirect (indr) | Used with applications that have a protocol next hop address that is remote. You are likely to see this next-hop type for internal BGP (IBGP) routes when the BGP next hop is a BGP neighbor that is not directly connected. |
| Interface | Used for a network address assigned to an interface. Unlike the router next hop, the interface next hop does not reference any specific node on the network. |
| Local (locl) | Local address on an interface. This next-hop type causes packets with this destination address to be received locally. |
| Multicast (mcst) | Wire multicast next hop (limited to the LAN). |
| Multicast discard (mdsc) | Multicast discard. |
| Multicast group (mgrp) | Multicast group member. |
| Receive (recv) | Receive. |
| Reject (rjct) | Discard. An ICMP unreachable message was sent. |
| Resolve (rslv) | Resolving next hop. |
| Routed multicast (mcrt) | Regular multicast next hop. |
| Router | <p>A specific node or set of nodes to which the routing device forwards packets that match the route prefix.</p> <p>To qualify as next-hop type router, the route must meet the following criteria:</p> <ul style="list-style-type: none"> • Must not be a direct or local subnet for the routing device. • Must have a next hop that is directly connected to the routing device. |
| Table | Routing table next hop. |
| Unicast (ucst) | Unicast. |
| Unilist (ulst) | List of unicast next hops. A packet sent to this next hop goes to any next hop in the list. |

Table 11 on page 125 describes all possible values for the **State** output field. A route can be in more than one state (for example, **<Active NoReadvrt Int Ext>**).

Table 11: State Output Field Values

| Value | Description |
|--|--|
| Accounting | Route needs accounting. |
| Active | Route is active. |
| Always Compare MED | Path with a lower multiple exit discriminator (MED) is available. |
| AS path | Shorter AS path is available. |
| Cisco Non-deterministic MED selection | Cisco nondeterministic MED is enabled, and a path with a lower MED is available. |
| Clone | Route is a clone. |
| Cluster list length | Length of cluster list sent by the route reflector. |
| Delete | Route has been deleted. |
| Ex | Exterior route. |
| Ext | BGP route received from an external BGP neighbor. |
| FlashAll | Forces all protocols to be notified of a change to any route, active or inactive, for a prefix. When not set, protocols are informed of a prefix only when the active route changes. |
| Hidden | Route not used because of routing policy. |
| IfCheck | Route needs forwarding RPF check. |
| IGP metric | Path through next hop with lower IGP metric is available. |
| Inactive reason | Flags for this route, which was not selected as best for a particular destination. |
| Initial | Route being added. |
| Int | Interior route. |
| Int Ext | BGP route received from an internal BGP peer or a BGP confederation peer. |
| Interior > Exterior > Exterior via Interior | Direct, static, IGP, or EBGp path is available. |
| Local Preference | Path with a higher local preference value is available. |
| Martian | Route is a martian (ignored because it is obviously invalid). |

Table 11: State Output Field Values (*continued*)

| Value | Description |
|---------------------------------------|---|
| MartianOK | Route exempt from martian filtering. |
| Next hop address | Path with lower metric next hop is available. |
| No difference | Path from neighbor with lower IP address is available. |
| NoReadvrt | Route not to be advertised. |
| NotBest | Route not chosen because it does not have the lowest MED. |
| Not Best in its group | Incoming BGP AS is not the best of a group (only one AS can be the best). |
| NotInstall | Route not to be installed in the forwarding table. |
| Number of gateways | Path with a greater number of next hops is available. |
| Origin | Path with a lower origin code is available. |
| Pending | Route pending because of a hold-down configured on another route. |
| Release | Route scheduled for release. |
| RIB preference | Route from a higher-numbered routing table is available. |
| Route Distinguisher | 64-bit prefix added to IP subnets to make them unique. |
| Route Metric or MED comparison | Route with a lower metric or MED is available. |
| Route Preference | Route with lower preference value is available |
| Router ID | Path through a neighbor with lower ID is available. |
| Secondary | Route not a primary route. |
| Unusable path | Path is not usable because of one of the following conditions: <ul style="list-style-type: none"> • The route is damped. • The route is rejected by an import policy. • The route is unresolved. |
| Update source | Last tiebreaker is the lowest IP address value. |

Table 12 on page 127 describes the possible values for the **Communities** output field.

Table 12: Communities Output Field Values

| Value | Description |
|---|---|
| <i>area-number</i> | 4 bytes, encoding a 32-bit area number. For AS-external routes, the value is 0 . A nonzero value identifies the route as internal to the OSPF domain, and as within the identified area. Area numbers are relative to a particular OSPF domain. |
| <i>bandwidth: local AS number:link-bandwidth-number</i> | Link-bandwidth community value used for unequal-cost load balancing. When BGP has several candidate paths available for multipath purposes, it does not perform unequal-cost load balancing according to the link-bandwidth community unless all candidate paths have this attribute. |
| <i>domain-id</i> | Unique configurable number that identifies the OSPF domain. |
| <i>domain-id-vendor</i> | Unique configurable number that further identifies the OSPF domain. |
| <i>link-bandwidth-number</i> | Link-bandwidth number: from 0 through 4,294,967,295 (bytes per second). |
| <i>local AS number</i> | Local AS number: from 1 through 65,535 . |
| <i>options</i> | 1 byte. Currently this is only used if the route type is 5 or 7 . Setting the least significant bit in the field indicates that the route carries a type 2 metric. |
| <i>origin</i> | (Used with VPNs) Identifies where the route came from. |
| <i>ospf-route-type</i> | 1 byte, encoded as 1 or 2 for intra-area routes (depending on whether the route came from a type 1 or a type 2 LSA); 3 for summary routes; 5 for external routes (area number must be 0); 7 for NSSA routes; or 129 for sham link endpoint addresses. |
| <i>route-type-vendor</i> | Displays the area number, OSPF route type, and option of the route. This is configured using the BGP extended community attribute 0x8000 . The format is area-number:ospf-route-type:options . |
| <i>rte-type</i> | Displays the area number, OSPF route type, and option of the route. This is configured using the BGP extended community attribute 0x0306 . The format is area-number:ospf-route-type:options . |
| <i>target</i> | Defines which VPN the route participates in; target has the format 32-bit IP address:16-bit number . For example, 10.19.0.0:100. |
| <i>unknown IANA</i> | Incoming IANA codes with a value between 0x1 and 0x7fff . This code of the BGP extended community attribute is accepted, but it is not recognized. |
| <i>unknown OSPF vendor community</i> | Incoming IANA codes with a value above 0x8000 . This code of the BGP extended community attribute is accepted, but it is not recognized. |

Sample Output

show route detail

```
user@host> show route detail

inet.0: 22 destinations, 23 routes (21 active, 0 holddown, 1 hidden)
10.10.0.0/16 (1 entry, 1 announced)
  *Static Preference: 5
    Next-hop reference count: 29
    Next hop: 192.168.71.254 via fxp0.0, selected
    State: <Active NoReadvrt Int Ext>
    Local AS: 69
    Age: 1:31:43
    Task: RT
    Announcement bits (2): 0-KRT 3-Resolve tree 2
    AS path: I

10.31.1.0/30 (2 entries, 1 announced)
  *Direct Preference: 0
    Next hop type: Interface
    Next-hop reference count: 2
    Next hop: via so-0/3/0.0, selected
    State: <Active Int>
    Local AS: 69
    Age: 1:30:17
    Task: IF
    Announcement bits (1): 3-Resolve tree 2
    AS path: I
  OSPF Preference: 10
    Next-hop reference count: 1
    Next hop: via so-0/3/0.0, selected
    State: <Int>
    Inactive reason: Route Preference
    Local AS: 69
    Age: 1:30:17 Metric: 1
    Area: 0.0.0.0
    Task: OSPF
    AS path: I

10.31.1.1/32 (1 entry, 1 announced)
  *Local Preference: 0
    Next hop type: Local
    Next-hop reference count: 7
    Interface: so-0/3/0.0
    State: <Active NoReadvrt Int>
    Local AS: 69
    Age: 1:30:20
    Task: IF
    Announcement bits (1): 3-Resolve tree 2
    AS path: I

...

10.31.2.0/30 (1 entry, 1 announced)
  *OSPF Preference: 10
    Next-hop reference count: 9
    Next hop: via so-0/3/0.0
    Next hop: 10.31.1.6 via ge-3/1/0.0, selected
    State: <Active Int>
    Local AS: 69
    Age: 1:29:56 Metric: 2
```

```

Area: 0.0.0.0
Task: OSPF
Announcement bits (2): 0-KRT 3-Resolve tree 2
AS path: I

...

224.0.0.2/32 (1 entry, 1 announced)
  *PIM    Preference: 0
          Next-hop reference count: 18
          State: <Active NoReadvrt Int>
          Local AS:    69
          Age: 1:31:45
          Task: PIM Recv
          Announcement bits (2): 0-KRT 3-Resolve tree 2
          AS path: I

...

224.0.0.22/32 (1 entry, 1 announced)
  *IGMP   Preference: 0
          Next-hop reference count: 18
          State: <Active NoReadvrt Int>
          Local AS:    69
          Age: 1:31:43
          Task: IGMP
          Announcement bits (2): 0-KRT 3-Resolve tree 2
          AS path: I

inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

10.255.70.103/32 (1 entry, 1 announced)
  State: <FlashAll>
  *RSVP   Preference: 7
          Next-hop reference count: 6
          Next hop: 10.31.1.6 via ge-3/1/0.0 weight 0x1, selected
          Label-switched-path green-r1-r3
          Label operation: Push 100096
          State: <Active Int>
          Local AS:    69
          Age: 1:25:49   Metric: 2
          Task: RSVP
          Announcement bits (2): 1-Resolve tree 1 2-Resolve tree 2
          AS path: I

10.255.71.238/32 (1 entry, 1 announced)
  State: <FlashAll>
  *RSVP   Preference: 7
          Next-hop reference count: 6
          Next hop: via so-0/3/0.0 weight 0x1, selected
          Label-switched-path green-r1-r2
          State: <Active Int>
          Local AS:    69
          Age: 1:25:49   Metric: 1
          Task: RSVP
          Announcement bits (2): 1-Resolve tree 1 2-Resolve tree 2
          AS path: I

private__inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

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

```

```

47.0005.80ff.f800.0000.0108.0001.0102.5507.1052/152 (1 entry, 0 announced)
    *Direct Preference: 0
        Next hop type: Interface
        Next-hop reference count: 1
        Next hop: via lo0.0, selected
        State: <Active Int>
        Local AS: 69
        Age: 1:31:44
        Task: IF
        AS path: I

mpls.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
0 (1 entry, 1 announced)
    *MPLS Preference: 0
        Next hop type: Receive
        Next-hop reference count: 6
        State: <Active Int>
        Local AS: 69
        Age: 1:31:45 Metric: 1
        Task: MPLS
        Announcement bits (1): 0-KRT
        AS path: I

...

mpls.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
299776 (1 entry, 1 announced)
TSI:
KRT in-kernel 299776 /52 -> {Flood}
    *RSVP Preference: 7
        Next hop type: Flood
        Next-hop reference count: 130
        Flood nexthop branches exceed maximum
        Address: 0x8ea65d0

...

800010 (1 entry, 1 announced)
    *VPLS Preference: 7
        Next-hop reference count: 2
        Next hop: via vt-3/2/0.32769, selected
        Label operation: Pop
        State: <Active Int>
        Age: 1:29:30
        Task: Common L2 VC
        Announcement bits (1): 0-KRT
        AS path: I

vt-3/2/0.32769 (1 entry, 1 announced)
    *VPLS Preference: 7
        Next-hop reference count: 2
        Next hop: 10.31.1.6 via ge-3/1/0.0 weight 0x1, selected
        Label-switched-path green-r1-r3
        Label operation: Push 800012, Push 100096(top)
        Protocol next hop: 10.255.70.103
        Push 800012
        Indirect next hop: 87272e4 1048574
        State: <Active Int>
        Age: 1:29:30 Metric2: 2
        Task: Common L2 VC
        Announcement bits (2): 0-KRT 1-Common L2 VC

```



```

AS path: I
Communities: target:11111:1 Layer2-info: encaps:VPLS,
control flags:, mtu: 0

inet6.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)

abcd::10:255:71:52/128 (1 entry, 0 announced)
  *Direct Preference: 0
    Next hop type: Interface
    Next-hop reference count: 1
    Next hop: via lo0.0, selected
    State: <Active Int>
    Local AS: 69
    Age: 1:31:44
    Task: IF
    AS path: I

fe80::280:42ff:fe10:f179/128 (1 entry, 0 announced)
  *Direct Preference: 0
    Next hop type: Interface
    Next-hop reference count: 1
    Next hop: via lo0.0, selected
    State: <Active NoReadvrt Int>
    Local AS: 69
    Age: 1:31:44
    Task: IF
    AS path: I

ff02::2/128 (1 entry, 1 announced)
  *PIM Preference: 0
    Next-hop reference count: 18
    State: <Active NoReadvrt Int>
    Local AS: 69
    Age: 1:31:45
    Task: PIM Recv6
    Announcement bits (1): 0-KRT
    AS path: I

ff02::d/128 (1 entry, 1 announced)
  *PIM Preference: 0
    Next-hop reference count: 18
    State: <Active NoReadvrt Int>
    Local AS: 69
    Age: 1:31:45
    Task: PIM Recv6
    Announcement bits (1): 0-KRT
    AS path: I

ff02::16/128 (1 entry, 1 announced)
  *MLD Preference: 0
    Next-hop reference count: 18
    State: <Active NoReadvrt Int>
    Local AS: 69
    Age: 1:31:43
    Task: MLD
    Announcement bits (1): 0-KRT
    AS path: I

private.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

fe80::280:42ff:fe10:f179/128 (1 entry, 0 announced)

```

```

*Direct Preference: 0
  Next hop type: Interface
  Next-hop reference count: 1
  Next hop: via lo0.16385, selected
  State: <Active NoReadvrt Int>
  Age: 1:31:44
  Task: IF
  AS path: I

green.l2vpn.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)

10.255.70.103:1:3:1/96 (1 entry, 1 announced)
  *BGP Preference: 170/-101
    Route Distinguisher: 10.255.70.103:1
    Next-hop reference count: 7
    Source: 10.255.70.103
    Protocol next hop: 10.255.70.103
    Indirect next hop: 2 no-forward
    State: <Secondary Active Int Ext>
    Local AS: 69 Peer AS: 69
    Age: 1:25:49 Metric2: 1
    AIGP 210
    Task: BGP_69.10.255.70.103+179
    Announcement bits (1): 0-green-l2vpn
    AS path: I
    Communities: target:11111:1 Layer2-info: encaps:VPLS,
    control flags:, mtu: 0
    Label-base: 800008, range: 8
    Localpref: 100
    Router ID: 10.255.70.103
    Primary Routing Table bgp.l2vpn.0

10.255.71.52:1:1:1/96 (1 entry, 1 announced)
  *L2VPN Preference: 170/-1
    Next-hop reference count: 5
    Protocol next hop: 10.255.71.52
    Indirect next hop: 0 -
    State: <Active Int Ext>
    Age: 1:31:40 Metric2: 1
    Task: green-l2vpn
    Announcement bits (1): 1-BGP.0.0.0.0+179
    AS path: I
    Communities: Layer2-info: encaps:VPLS, control flags:Site-Down,
    mtu: 0
    Label-base: 800016, range: 8, status-vector: 0x9F

10.255.71.52:1:5:1/96 (1 entry, 1 announced)
  *L2VPN Preference: 170/-101
    Next-hop reference count: 5
    Protocol next hop: 10.255.71.52
    Indirect next hop: 0 -
    State: <Active Int Ext>
    Age: 1:31:40 Metric2: 1
    Task: green-l2vpn
    Announcement bits (1): 1-BGP.0.0.0.0+179
    AS path: I
    Communities: Layer2-info: encaps:VPLS, control flags:, mtu: 0
    Label-base: 800008, range: 8, status-vector: 0x9F

...

```

```

12circuit.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
10.245.255.63:CtrlWord:4:3:Local/96 (1 entry, 1 announced)
  *L2CKT Preference: 7
    Next hop: via so-1/1/2.0 weight 1, selected
    Label-switched-path my-lsp
    Label operation: Push 100000[0]
    Protocol next hop: 10.245.255.63 Indirect next hop: 86af000 296
    State: <Active Int>
    Local AS: 99
    Age: 10:21
    Task: 12 circuit
    Announcement bits (1): 0-LDP
    AS path: I
    VC Label 100000, MTU 1500, VLAN ID 512

```

show route detail (with BGP Multipath)

```
user@host> show route detail
```

```

10.1.1.8/30 (2 entries, 1 announced)
  *BGP Preference: 170/-101
    Next hop type: Router, Next hop index: 262142
    Address: 0x901a010
    Next-hop reference count: 2
    Source: 10.1.1.2
    Next hop: 10.1.1.2 via ge-0/3/0.1, selected
    Next hop: 10.1.1.6 via ge-0/3/0.5
    State: <Active Ext>
    Local AS: 1 Peer AS: 2
    Age: 5:04:43
    Task: BGP_2.10.1.1.2+59955
    Announcement bits (1): 0-KRT
    AS path: 2 I
    Accepted Multipath
    Localpref: 100
    Router ID: 1.1.1.2
  BGP Preference: 170/-101
    Next hop type: Router, Next hop index: 678
    Address: 0x8f97520
    Next-hop reference count: 9
    Source: 10.1.1.6
    Next hop: 10.1.1.6 via ge-0/3/0.5, selected
    State: <NotBest Ext>
    Inactive reason: Not Best in its group - Active preferred
    Local AS: 1 Peer AS: 2
    Age: 5:04:43
    Task: BGP_2.10.1.1.6+58198
    AS path: 2 I
    Accepted MultipathContrib
    Localpref: 100
    Router ID: 1.1.1.3

```

show route forwarding-table

| | |
|-----------------------------------|---|
| Syntax | <pre>show route forwarding-table <detail extensive summary> <all> <ccc interface-name> <destination destination-prefix> <family family matching matching> <interface-name interface-name> <label name> <matching matching> <multicast> <table (default logical-system-name/routing-instance-name routing-instance-name)> <vlan (all vlan-name)> <vpn vpn></pre> |
| Syntax (MX Series Routers) | <pre>show route forwarding-table <detail extensive summary> <all> <bridge-domain (all domain-name)> <ccc interface-name> <destination destination-prefix> <family family matching matching> <interface-name interface-name> <label name> <learning-vlan-id learning-vlan-id> <matching matching> <multicast> <table (default logical-system-name/routing-instance-name routing-instance-name)> <vlan (all vlan-name)> <vpn vpn></pre> |
| Syntax (Routing Matrix) | <pre>show route forwarding-table <detail extensive summary> <all> <ccc interface-name> <destination destination-prefix> <family family matching matching> <interface-name interface-name> <matching matching> <label name> <lcc number> <multicast> <table routing-instance-name> <vpn vpn></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Option bridge-domain introduced in Junos OS Release 7.5</p> <p>Option learning-vlan-id introduced in Junos OS Release 8.4</p> <p>Options all and vlan introduced in Junos OS Release 9.6.</p> <p>Command introduced in Junos OS Release 11.3 for the QFX Series.</p> |

Description Display the Routing Engine's forwarding table, including the network-layer prefixes and their next hops. This command is used to help verify that the routing protocol process has relayed the correction information to the forwarding table. The Routing Engine constructs and maintains one or more routing tables. From the routing tables, the Routing Engine derives a table of active routes, called the forwarding table.



NOTE: The Routing Engine copies the forwarding table to the Packet Forwarding Engine, the part of the router that is responsible for forwarding packets. To display the entries in the Packet Forwarding Engine's forwarding table, use the `show pfe route` command.

Options **none**—Display the routes in the forwarding tables. By default, the `show route forwarding-table` command does not display information about private, or internal, forwarding tables.

detail | extensive | summary—(Optional) Display the specified level of output.

all—(Optional) Display routing table entries for all forwarding tables, including private, or internal, tables.

bridge-domain (all | *bridge-domain-name*)—(MX Series routers only) (Optional) Display route entries for all bridge domains or the specified bridge domain.

ccc *interface-name*—(Optional) Display route entries for the specified circuit cross-connect interface.

destination *destination-prefix*—(Optional) Destination prefix.

family *family*—(Optional) Display routing table entries for the specified family: **fibre-channel**, **fmembers**, **inet**, **inet6**, **iso**, **mpls**, **tnp**, **unix**, **vpls**, or **vlan-classification**.

interface-name *interface-name*—(Optional) Display routing table entries for the specified interface.

label *name*—(Optional) Display route entries for the specified label.

lcc *number*—(Routing Matrix only) (Optional) On a routing matrix composed of a TX Matrix Plus router and T640 routers configured in the routing matrix, display information for the specified T640 router (or line-card chassis) connected to the TX Matrix router. On a routing matrix composed of the TX Matrix Plus router and T1600 routers configured in the routing matrix, display information for the specified T1600 router (or line-card chassis) connected to the TX Matrix Plus router. Replace ***number*** with a value from 0 through 3.

learning-vlan-id *learning-vlan-id*—(MX Series routers only) (Optional) Display learned information for all VLANs or for the specified VLAN.

matching *matching*—(Optional) Display routing table entries matching the specified prefix or prefix length.

multicast—(Optional) Display routing table entries for multicast routes.

table (**default** | *logical-system-name/routing-instance-name* |

routing-instance-name)—(Optional) Display route entries for all the routing tables in the main routing instance or for the specified routing instance. If your device supports logical systems, you can also display route entries for the specified logical system and routing instance. To view the routing instances on your device, use the **show route instance** command.

vlan (**all** | *vlan-name*)—(Optional) Display information for all VLANs or for the specified VLAN.

vpn *vpn*—(Optional) Display routing table entries for a specified VPN.

Required Privilege Level view

List of Sample Output [show route forwarding-table on page 139](#)
[show route forwarding-table detail on page 139](#)
[show route forwarding-table destination extensive \(Weights and Balances\) on page 140](#)
[show route forwarding-table extensive on page 141](#)
[show route forwarding-table extensive \(RPF\) on page 142](#)
[show route forwarding-table family mpls on page 144](#)
[show route forwarding-table family vpls on page 144](#)
[show route forwarding-table family vpls extensive on page 144](#)
[show route forwarding-table table default on page 145](#)
[show route forwarding-table table](#)
[logical-system-name/routing-instance-name on page 146](#)
[show route forwarding-table vpn on page 147](#)

Output Fields [Table 13 on page 136](#) lists the output fields for the **show route forwarding-table** command. Output fields are listed in the approximate order in which they appear. Field names might be abbreviated (as shown in parentheses) when no level of output is specified, or when the **detail** keyword is used instead of the **extensive** keyword.

Table 13: show route forwarding-table Output Fields

| Field Name | Field Description | Level of Output |
|-----------------------|--|-------------------------|
| Logical system | Name of the logical system. This field is displayed if you specify the table <i>logical-system-name/routing-instance-name</i> option on a device that is configured for and supports logical systems. | All levels |
| Routing table | Name of the routing table (for example, inet, inet6, mpls). | All levels |
| Address family | Address family (for example, IP, IPv6, ISO, MPLS, and VPLS). | All levels |
| Destination | Destination of the route. | detail extensive |

Table 13: show route forwarding-table Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|--------------------------------|--|-------------------------|
| Route Type (Type) | How the route was placed into the forwarding table. When the detail keyword is used, the route type might be abbreviated (as shown in parentheses): <ul style="list-style-type: none"> • cloned (clon)—(TCP or multicast only) Cloned route. • destination (dest)—Remote addresses directly reachable through an interface. • destination down (iddn)—Destination route for which the interface is unreachable. • interface cloned (ifcl)—Cloned route for which the interface is unreachable. • route down (ifdn)—Interface route for which the interface is unreachable. • ignore (ignr)—Ignore this route. • interface (intf)—Installed as a result of configuring an interface. • permanent (perm)—Routes installed by the kernel when the routing table is initialized. • user—Routes installed by the routing protocol process or as a result of the configuration. | All levels |
| Route Reference (RtRef) | Number of routes to reference. | detail extensive |
| Flags | Route type flags: <ul style="list-style-type: none"> • none—No flags are enabled. • accounting—Route has accounting enabled. • cached—Cache route. • incoming-iface interface-number—Check against incoming interface. • prefix load balance—Load balancing is enabled for this prefix. • rt nh decoupled—Route has been decoupled from the next hop to the destination. • sent to PFE—Route has been sent to the Packet Forwarding Engine. • static—Static route. | extensive |
| Next hop | IP address of the next hop to the destination. | detail extensive |

Table 13: show route forwarding-table Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|-----------------------------------|--|------------------------------|
| Next hop Type (Type) | <p>Next-hop type. When the detail keyword is used, the next-hop type might be abbreviated (as indicated in parentheses):</p> <ul style="list-style-type: none"> • broadcast (bcst)—Broadcast. • deny—Deny. • discard (dscd)—Discard. • hold—Next hop is waiting to be resolved into a unicast or multicast type. • indexed (idxd)—Indexed next hop. • indirect (indr)—Indirect next hop. • local (locl)—Local address on an interface. • routed multicast (mcrst)—Regular multicast next hop. • multicast (mcst)—Wire multicast next hop (limited to the LAN). • multicast discard (mdsc)—Multicast discard. • multicast group (mgrp)—Multicast group member. • receive (rcv)—Receive. • reject (rjct)—Discard. An ICMP unreachable message was sent. • resolve (rslv)—Resolving the next hop. • unicast (ucst)—Unicast. • unilist (ulst)—List of unicast next hops. A packet sent to this next hop goes to any next hop in the list. | detail extensive |
| Index | Software index of the next hop that is used to route the traffic for a given prefix. | detail extensive none |
| Route interface-index | Logical interface index from which the route is learned. For example, for interface routes, this is the logical interface index of the route itself. For static routes, this field is zero. For routes learned through routing protocols, this is the logical interface index from which the route is learned. | extensive |
| Reference (NhRef) | Number of routes that refer to this next hop. | detail extensive none |
| Next-hop interface (Netif) | Interface used to reach the next hop. | detail extensive none |
| Weight | Value used to distinguish primary, secondary, and fast reroute backup routes. Weight information is available when MPLS label-switched path (LSP) link protection, node-link protection, or fast reroute is enabled, or when the standby state is enabled for secondary paths. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible (see the Balance field description). | extensive |
| Balance | Balance coefficient indicating how traffic of unequal cost is distributed among next hops when a router is performing unequal-cost load balancing. This information is available when you enable BGP multipath load balancing. | extensive |
| RPF interface | List of interfaces from which the prefix can be accepted. Reverse path forwarding (RPF) information is displayed only when rpf-check is configured on the interface. | extensive |

Sample Output

```
show route
forwarding-table
```

```
user@host> show route forwarding-table
```

```
Routing table: default.inet
```

```
Internet:
```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|------------------|------|-------|-------------------|------|-------|-------|------------|
| default | perm | 0 | | rjct | 46 | 4 | |
| 0.0.0.0/32 | perm | 0 | | dscd | 44 | 1 | |
| 1.1.1.0/24 | ifdn | 0 | | rslv | 608 | 1 | ge-2/0/1.0 |
| 1.1.1.0/32 | iddn | 0 | 1.1.1.0 | recv | 606 | 1 | ge-2/0/1.0 |
| 1.1.1.1/32 | user | 0 | | rjct | 46 | 4 | |
| 1.1.1.1/32 | intf | 0 | 1.1.1.1 | loc1 | 607 | 2 | |
| 1.1.1.1/32 | iddn | 0 | 1.1.1.1 | loc1 | 607 | 2 | |
| 1.1.1.255/32 | iddn | 0 | ff:ff:ff:ff:ff:ff | bcst | 605 | 1 | ge-2/0/1.0 |
| 10.0.0.0/24 | intf | 0 | | rslv | 616 | 1 | ge-2/0/0.0 |
| 10.0.0.0/32 | dest | 0 | 10.0.0.0 | recv | 614 | 1 | ge-2/0/0.0 |
| 10.0.0.1/32 | intf | 0 | 10.0.0.1 | loc1 | 615 | 2 | |
| 10.0.0.1/32 | dest | 0 | 10.0.0.1 | loc1 | 615 | 2 | |
| 10.0.0.255/32 | dest | 0 | 10.0.0.255 | bcst | 613 | 1 | ge-2/0/0.0 |
| 10.1.1.0/24 | ifdn | 0 | | rslv | 612 | 1 | ge-2/0/1.0 |
| 10.1.1.0/32 | iddn | 0 | 10.1.1.0 | recv | 610 | 1 | ge-2/0/1.0 |
| 10.1.1.1/32 | user | 0 | | rjct | 46 | 4 | |
| 10.1.1.1/32 | intf | 0 | 10.1.1.1 | loc1 | 611 | 2 | |
| 10.1.1.1/32 | iddn | 0 | 10.1.1.1 | loc1 | 611 | 2 | |
| 10.1.1.255/32 | iddn | 0 | ff:ff:ff:ff:ff:ff | bcst | 609 | 1 | ge-2/0/1.0 |
| 10.209.0.0/16 | user | 0 | 10.209.63.254 | ucst | 419 | 20 | fxp0.0 |
| 10.209.0.0/16 | user | 1 | 0:12:1e:ca:98:0 | ucst | 419 | 20 | fxp0.0 |
| 10.209.0.0/18 | intf | 0 | | rslv | 418 | 1 | fxp0.0 |
| 10.209.0.0/32 | dest | 0 | 10.209.0.0 | recv | 416 | 1 | fxp0.0 |
| 10.209.2.131/32 | intf | 0 | 10.209.2.131 | loc1 | 417 | 2 | |
| 10.209.2.131/32 | dest | 0 | 10.209.2.131 | loc1 | 417 | 2 | |
| 10.209.17.55/32 | dest | 0 | 0:30:48:5b:78:d2 | ucst | 435 | 1 | fxp0.0 |
| 10.209.63.42/32 | dest | 0 | 0:23:7d:58:92:ca | ucst | 434 | 1 | fxp0.0 |
| 10.209.63.254/32 | dest | 0 | 0:12:1e:ca:98:0 | ucst | 419 | 20 | fxp0.0 |
| 10.209.63.255/32 | dest | 0 | 10.209.63.255 | bcst | 415 | 1 | fxp0.0 |
| 10.227.0.0/16 | user | 0 | 10.209.63.254 | ucst | 419 | 20 | fxp0.0 |

```
...
```

```
Routing table: iso
```

```
ISO:
```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|--|------|-------|-----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 27 | 1 | |
| 47.0005.80ff.f800.0000.0108.0003.0102.5524.5220.00 | intf | 0 | loc1 28 1 | | | | |

```
Routing table: inet6
```

```
Internet6:
```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 6 | 1 | |
| ff00::/8 | perm | 0 | | mdsc | 4 | 1 | |
| ff02::1/128 | perm | 0 | ff02::1 | mcst | 3 | 1 | |

```
Routing table: ccc
```

```
MPLS:
```

| Interface.Label | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-----------------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 16 | 1 | |
| 100004(top)fe-0/0/1.0 | | | | | | | |

show route forwarding-table detail

```
user@host> show route forwarding-table detail
```

```
Routing table: inet
```

```
Internet:
```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-----------------|------|-------|------------------|------|-------|-------|------------|
| default | user | 2 | 0:90:69:8e:b1:1b | ucst | 132 | 4 | fxp0.0 |
| default | perm | 0 | | rjct | 14 | 1 | |
| 10.1.1.0/24 | intf | 0 | ff.3.0.21 | ucst | 322 | 1 | so-5/3/0.0 |
| 10.1.1.0/32 | dest | 0 | 10.1.1.0 | recv | 324 | 1 | so-5/3/0.0 |
| 10.1.1.1/32 | intf | 0 | 10.1.1.1 | loc1 | 321 | 1 | |
| 10.1.1.255/32 | dest | 0 | 10.1.1.255 | bcst | 323 | 1 | so-5/3/0.0 |
| 10.21.21.0/24 | intf | 0 | ff.3.0.21 | ucst | 326 | 1 | so-5/3/0.0 |
| 10.21.21.0/32 | dest | 0 | 10.21.21.0 | recv | 328 | 1 | so-5/3/0.0 |
| 10.21.21.1/32 | intf | 0 | 10.21.21.1 | loc1 | 325 | 1 | |
| 10.21.21.255/32 | dest | 0 | 10.21.21.255 | bcst | 327 | 1 | so-5/3/0.0 |
| 127.0.0.1/32 | intf | 0 | 127.0.0.1 | loc1 | 320 | 1 | |
| 172.17.28.19/32 | clon | 1 | 192.168.4.254 | ucst | 132 | 4 | fxp0.0 |
| 172.17.28.44/32 | clon | 1 | 192.168.4.254 | ucst | 132 | 4 | fxp0.0 |

```
...
```

```
Routing table: private1__inet
```

```
Internet:
```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|--------|
| default | perm | 0 | | rjct | 46 | 1 | |
| 10.0.0.0/8 | intf | 0 | | rslv | 136 | 1 | fxp1.0 |
| 10.0.0.0/32 | dest | 0 | 10.0.0.0 | recv | 134 | 1 | fxp1.0 |
| 10.0.0.4/32 | intf | 0 | 10.0.0.4 | loc1 | 135 | 2 | |
| 10.0.0.4/32 | dest | 0 | 10.0.0.4 | loc1 | 135 | 2 | |

```
...
```

```
Routing table: iso
```

```
ISO:
```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 38 | 1 | |

```
Routing table: inet6
```

```
Internet6:
```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 22 | 1 | |
| ff00::/8 | perm | 0 | | mdsc | 21 | 1 | |
| ff02::1/128 | perm | 0 | ff02::1 | mcst | 17 | 1 | |

```
...
```

```
Routing table: mpls
```

```
MPLS:
```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 28 | 1 | |

show route forwarding-table destination extensive

```
user@host> show route forwarding-table destination 3.4.2.1 extensive
```

```
Routing table: inet [Index 0]
```

```
Internet:
```

(Weights and Balances)

```

Destination: 3.4.2.1/32
Route type: user
Route reference: 0
Flags: sent to PFE
Next-hop type: unicast
Nexthop: 4.4.4.4
Next-hop type: unicast
Next-hop interface: so-1/1/0.0
Nexthop: 145.12.1.2
Next-hop type: unicast
Next-hop interface: so-0/1/2.0

Route interface-index: 0
Index: 262143 Reference: 1
Index: 335 Reference: 2
Weight: 22 Balance: 3
Index: 337 Reference: 2
Weight: 33 Balance: 33

```

show route forwarding-table extensive

```

user@host> show route forwarding-table extensive
Routing table: inet [Index 0]
Internet:

Destination: default
Route type: user
Route reference: 2
Flags: sent to PFE
Nexthop: 0:90:69:8e:b1:1b
Next-hop type: unicast
Next-hop interface: fxp0.0

Route interface-index: 0
Index: 132 Reference: 4

Destination: default
Route type: permanent
Route reference: 0
Flags: none
Next-hop type: reject

Route interface-index: 0
Index: 14 Reference: 1

Destination: 127.0.0.1/32
Route type: interface
Route reference: 0
Flags: sent to PFE
Nexthop: 127.0.0.1
Next-hop type: local

Route interface-index: 0
Index: 320 Reference: 1

...

Routing table: private1__inet [Index 1]
Internet:

Destination: default
Route type: permanent
Route reference: 0
Flags: sent to PFE
Next-hop type: reject

Route interface-index: 0
Index: 46 Reference: 1

Destination: 10.0.0.0/8
Route type: interface
Route reference: 0
Flags: sent to PFE
Next-hop type: resolve
Next-hop interface: fxp1.0

Route interface-index: 3
Index: 136 Reference: 1

...

Routing table: iso [Index 0]
ISO:

Destination: default

```

```

Route type: permanent
Route reference: 0
Flags: sent to PFE
Next-hop type: reject
Route interface-index: 0
Index: 38      Reference: 1

Routing table: inet6 [Index 0]
Internet6:

Destination: default
Route type: permanent
Route reference: 0
Flags: sent to PFE
Next-hop type: reject
Route interface-index: 0
Index: 22      Reference: 1

Destination: ff00::/8
Route type: permanent
Route reference: 0
Flags: sent to PFE
Next-hop type: multicast discard
Route interface-index: 0
Index: 21      Reference: 1

...

Routing table: private1__inet6 [Index 1]
Internet6:

Destination: default
Route type: permanent
Route reference: 0
Flags: sent to PFE
Next-hop type: reject
Route interface-index: 0
Index: 54      Reference: 1

Destination: fe80::2a0:a5ff:fe3d:375/128
Route type: interface
Route reference: 0
Flags: sent to PFE
Nexthop: fe80::2a0:a5ff:fe3d:375
Next-hop type: local
Route interface-index: 0
Index: 75      Reference: 1

...

```

show route forwarding-table extensive (RPF)

The next example is based on the following configuration, which enables an RPF check on all routes that are learned from this interface, including the interface route:

```

so-1/1/0 {
  unit 0 {
    family inet {
      rpf-check;
      address 15.95.1.2/30;
    }
  }
}

```

```

user@host> show route forwarding-table extensive
Routing table: inet [Index 0]
Internet:
...
...
Destination: 15.95.1.3/32
Route type: destination
Route reference: 0
Route interface-index: 67

```

```
Flags: sent to PFE
Nexthop: 15.95.1.3
Next-hop type: broadcast      Index: 328      Reference: 1
Next-hop interface: so-1/1/0.0
RPF interface: so-1/1/0.0
```

show route forwarding-table family mpls

```

user@host> show route forwarding-table family mpls
Routing table: mpls
MPLS:
Destination      Type RtRef Next hop      Type Index NhRef Netif
default          perm  0          Type Index NhRef Netif
0                user  0          rjct   19    1
1                user  0          rcv    18    3
2                user  0          rcv    18    3
100000           user  0 10.31.1.6      swap  100001 fe-1/1/0.0
800002           user  0          Pop                    vt-0/3/0.32770

vt-0/3/0.32770 (VPLS)
                  user  0          indr   351    4
                  Push 800000, Push 100002(top)

so-0/0/0.0

```

show route forwarding-table family vpls

```

user@host> show route forwarding-table family vpls
Routing table: green.vpls
VPLS:
Destination      Type RtRef Next hop      Type Index NhRef Netif
default          dynm  0          flood  353    1
default          perm  0          rjct   298    1
fe-0/1/0.0       dynm  0          flood  355    1
00:90:69:0c:20:1f/48
                  <<<<<<Remote CE
                  dynm  0          indr   351    4
                  Push 800000, Push 100002(top)

so-0/0/0.0
00:90:69:85:b0:1f/48
                  <<<<<<Local CE
                  dynm  0          ucst   354    2 fe-0/1/0.0

```

show route forwarding-table family vpls extensive

```

user@host> show route forwarding-table family vpls extensive
Routing table: green.vpls [Index 2]
VPLS:

Destination: default
Route type: dynamic
Route reference: 0
Flags: sent to PFE
Next-hop type: flood
Next-hop type: unicast
Next-hop interface: fe-0/1/3.0
Next-hop type: unicast
Next-hop interface: fe-0/1/2.0
Route interface-index: 72
Index: 289 Reference: 1
Index: 291 Reference: 3
Index: 290 Reference: 3

Destination: default
Route type: permanent
Route reference: 0
Flags: none
Next-hop type: discard
Route interface-index: 0
Index: 341 Reference: 1

Destination: fe-0/1/2.0
Route type: dynamic
Route reference: 0
Flags: sent to PFE
Next-hop type: flood
Next-hop type: indirect
Next-hop type: Push 800016
Route interface-index: 69
Index: 293 Reference: 1
Index: 363 Reference: 4

```

```

Next-hop interface: at-1/0/1.0
Next-hop type: indirect          Index: 301      Reference: 5
Next hop: 10.31.3.2
Next-hop type: Push 800000
Next-hop interface: fe-0/1/1.0
Next-hop type: unicast          Index: 291      Reference: 3
Next-hop interface: fe-0/1/3.0

Destination: fe-0/1/3.0
Route type: dynamic
Route reference: 0              Route interface-index: 70
Flags: sent to PFE
Next-hop type: flood           Index: 292      Reference: 1
Next-hop type: indirect        Index: 363      Reference: 4
Next-hop type: Push 800016
Next-hop interface: at-1/0/1.0
Next-hop type: indirect        Index: 301      Reference: 5
Next hop: 10.31.3.2
Next-hop type: Push 800000
Next-hop interface: fe-0/1/1.0
Next-hop type: unicast          Index: 290      Reference: 3
Next-hop interface: fe-0/1/2.0

Destination: 10:00:00:01:01:01/48
Route type: dynamic
Route reference: 0              Route interface-index: 70
Flags: sent to PFE, prefix load balance
Next-hop type: unicast          Index: 291      Reference: 3
Next-hop interface: fe-0/1/3.0
Route used as destination:
  Packet count:      6640    Byte count:      675786
Route used as source
  Packet count:      6894    Byte count:      696424

Destination: 10:00:00:01:01:04/48
Route type: dynamic
Route reference: 0              Route interface-index: 69
Flags: sent to PFE, prefix load balance
Next-hop type: unicast          Index: 290      Reference: 3
Next-hop interface: fe-0/1/2.0
Route used as destination:
  Packet count:      96     Byte count:      8079
Route used as source:
  Packet count:      296    Byte count:      24955

Destination: 10:00:00:01:03:05/48
Route type: dynamic
Route reference: 0              Route interface-index: 74
Flags: sent to PFE, prefix load balance
Next-hop type: indirect        Index: 301      Reference: 5
Next hop: 10.31.3.2
Next-hop type: Push 800000
Next-hop interface: fe-0/1/1.0

```

show route forwarding-table table default

```

user@host> show route forwarding-table table default
Routing table: default.inet

```

```

Internet:
Destination      Type RtRef Next hop      Type Index NhRef Netif
default          perm  0
0.0.0.0/32       perm  0
10.0.60.0/30     user  0 10.0.60.13  ucst  713  5 fe-0/1/3.0

```

```

10.0.60.12/30      intf      0                rslv    688      1 fe-0/1/3.0
10.0.60.12/32      dest      0 10.0.60.12      recv    686      1 fe-0/1/3.0
10.0.60.13/32      dest      0 0:5:85:8b:bc:22 ucst    713      5 fe-0/1/3.0
10.0.60.14/32      intf      0 10.0.60.14      locl    687      2
10.0.60.14/32      dest      0 10.0.60.14      locl    687      2
10.0.60.15/32      dest      0 10.0.60.15      bcst    685      1 fe-0/1/3.0
10.0.67.12/30      user      0 10.0.60.13      ucst    713      5 fe-0/1/3.0
10.0.80.0/30       ifdn      0 ff.3.0.21       ucst    676      1 so-0/0/1.0
10.0.80.0/32      dest      0 10.0.80.0       recv    678      1 so-0/0/1.0
10.0.80.2/32       user      0                rjct     36      2
10.0.80.2/32      intf      0 10.0.80.2       locl    675      1
10.0.80.3/32      dest      0 10.0.80.3       bcst    677      1 so-0/0/1.0
10.0.90.12/30      intf      0                rslv    684      1 fe-0/1/0.0
10.0.90.12/32      dest      0 10.0.90.12      recv    682      1 fe-0/1/0.0
10.0.90.14/32      intf      0 10.0.90.14      locl    683      2
10.0.90.14/32      dest      0 10.0.90.14      locl    683      2
10.0.90.15/32      dest      0 10.0.90.15      bcst    681      1 fe-0/1/0.0
10.5.0.0/16        user      0 192.168.187.126 ucst    324     15 fxp0.0
10.10.0.0/16        user      0 192.168.187.126 ucst    324     15 fxp0.0
10.13.10.0/23       user      0 192.168.187.126 ucst    324     15 fxp0.0
10.84.0.0/16        user      0 192.168.187.126 ucst    324     15 fxp0.0
10.150.0.0/16       user      0 192.168.187.126 ucst    324     15 fxp0.0
10.157.64.0/19      user      0 192.168.187.126 ucst    324     15 fxp0.0
10.209.0.0/16       user      0 192.168.187.126 ucst    324     15 fxp0.0

```

...

Routing table: default.iso

ISO:

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 60 | 1 | |

Routing table: default.inet6

Internet6:

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 44 | 1 | |
| ::/128 | perm | 0 | | dscd | 42 | 1 | |
| ff00::/8 | perm | 0 | | mdsc | 43 | 1 | |
| ff02::1/128 | perm | 0 | ff02::1 | mcst | 39 | 1 | |

Routing table: default.mpls

MPLS:

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | dscd | 50 | 1 | |

show route
forwarding-table table
logical-system-name

user@host> show route forwarding-table table R4/vpn-red

Logical system: R4

Routing table: vpn-red.inet

Internet:

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|--------------|------|-------|--|------|-------|-------|------------|
| default | perm | 0 | | rjct | 563 | 1 | |
| 0.0.0.0/32 | perm | 0 | | dscd | 561 | 2 | |
| 1.0.0.1/32 | user | 0 | | dscd | 561 | 2 | |
| 2.0.2.0/24 | intf | 0 | | rslv | 771 | 1 | ge-1/2/0.3 |
| 2.0.2.0/32 | dest | 0 | 2.0.2.0 | recv | 769 | 1 | ge-1/2/0.3 |
| 2.0.2.1/32 | intf | 0 | 2.0.2.1 | locl | 770 | 2 | |
| 2.0.2.1/32 | dest | 0 | 2.0.2.1 | locl | 770 | 2 | |
| 2.0.2.2/32 | dest | 0 | 0.4.80.3.0.1b.c0.d5.e4.bd.0.1b.c0.d5.e4.bc.8.0 | ucst | 789 | 1 | ge-1/2/0.3 |
| 2.0.2.255/32 | dest | 0 | 2.0.2.255 | bcst | 768 | 1 | ge-1/2/0.3 |
| 224.0.0.0/4 | perm | 1 | | mdsc | 562 | 1 | |


```

224.0.0.1/32      perm      0 224.0.0.1      mcst    558      1
255.255.255.255/32 perm      0                bcst    559      1

```

```

Logical system: R4
Routing table: vpn-red.iso
ISO:

```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 608 | 1 | |

```

Logical system: R4
Routing table: vpn-red.inet6
Internet6:

```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 708 | 1 | |
| ::/128 | perm | 0 | | dscd | 706 | 1 | |
| ff00::/8 | perm | 0 | | mdsc | 707 | 1 | |
| ff02::1/128 | perm | 0 | ff02::1 | mcst | 704 | 1 | |

```

Logical system: R4
Routing table: vpn-red.mpls
MPLS:

```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | dscd | 638 | | |

show route
forwarding-table vpn

```

user@host> show route forwarding-table vpn VPN-A
Routing table:: VPN-A.inet

```

```

Internet:
Destination      Type RtRef Nexthop          Type Index NhRef Netif
default          perm  0                rjct   4    4
10.39.10.20/30   intf  0 ff.3.0.21          ucst   40    1
so-0/0/0.0
10.39.10.21/32   intf  0 10.39.10.21        locl   36    1
10.255.14.172/32 user   0                ucst   69    2
so-0/0/0.0
10.255.14.175/32 user   0                indr   81    3
Push 100004, Push

100004(top) so-1/0/0.0
224.0.0.0/4      perm  2                mdsc   5    3
224.0.0.1/32     perm  0 224.0.0.1          mcst   1    8
224.0.0.5/32     user   1 224.0.0.5          mcst   1    8
255.255.255.255/32 perm  0                bcst   2    3

```

show route hidden

| | |
|---------------------------------|--|
| Syntax | <code>show route hidden</code> <code><brief detail extensive terse></code> <code><logical-system (all <i>logical-system-name</i>)></code> |
| Release Information | Command introduced before Junos OS Release 7.4. |
| Description | Display only hidden route information. A hidden route is unusable, even if it is the best path. |
| Options | brief detail extensive terse —(Optional) Display the specified level of output. If you do not specify a level of output, the system defaults to brief . logical-system (all <i>logical-system-name</i>) —(Optional) Perform this operation on all logical systems or on a particular logical system. |
| Required Privilege Level | view |
| List of Sample Output | show route hidden on page 149 show route hidden detail on page 149 show route hidden extensive on page 150 show route hidden terse on page 150 |
| Output Fields | For information about output fields, see the output field table for the show route command, the show route detail command, the show route extensive command, or the show route terse command. |

Sample Output

show route hidden

```

user@host> show route hidden
inet.0: 25 destinations, 26 routes (24 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
127.0.0.1/32      [Direct/0] 04:26:38
                  > via lo0.0

private1___.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

red.inet.0: 6 destinations, 8 routes (4 active, 0 holddown, 3 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
10.5.5.5/32      [BGP/170] 03:44:10, localpref 100, from 10.4.4.4
                  AS path: 100 I
                  Unusable
10.12.1.0/24     [BGP/170] 03:44:10, localpref 100, from 10.4.4.4
                  AS path: 100 I
                  Unusable
10.12.80.4/30    [BGP/170] 03:44:10, localpref 100, from 10.4.4.4
                  AS path: I
                  Unusable
...

```

show route hidden detail

```

user@host> show route hidden detail

inet.0: 25 destinations, 26 routes (24 active, 0 holddown, 1 hidden)
Restart Complete
127.0.0.1/32 (1 entry, 0 announced)
    Direct Preference: 0
        Next hop type: Interface
        Next-hop reference count: 1
        Next hop: via lo0.0, selected
        State: <Hidden Martian Int>
        Local AS: 1
        Age: 4:27:37
        Task: IF
        AS path: I

private1___.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

red.inet.0: 6 destinations, 8 routes (4 active, 0 holddown, 3 hidden)
Restart Complete
10.5.5.5/32 (1 entry, 0 announced)
    BGP Preference: 170/-101
        Route Distinguisher: 10.4.4.4:4
        Next hop type: Unusable
        Next-hop reference count: 6
        State: <Secondary Hidden Int Ext>
        Local AS: 1 Peer AS: 1
        Age: 3:45:09
        Task: BGP_1.10.4.4.4+2493
        AS path: 100 I
        Communities: target:1:999
        VPN Label: 100064
        Localpref: 100
        Router ID: 10.4.4.4

```

Primary Routing Table bgp.13vpn.0

...

show route hidden extensive

The output for the **show route hidden extensive** command is identical to that of the **show route hidden detail** command. For sample output, see [show route hidden detail on page 149](#).

show route hidden terse

```
user@host> show route hidden terse
```

```
inet.0: 25 destinations, 26 routes (24 active, 0 holddown, 1 hidden)
```

```
Restart Complete
```

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

| A Destination | P Prf | Metric 1 | Metric 2 | Next hop | AS path |
|---------------|-------|----------|----------|----------|---------|
| 127.0.0.1/32 | D 0 | | | >1o0.0 | |

```
private1___.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
```

```
red.inet.0: 6 destinations, 8 routes (4 active, 0 holddown, 3 hidden)
```

```
Restart Complete
```

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

| A Destination | P Prf | Metric 1 | Metric 2 | Next hop | AS path |
|---------------|-------|----------|----------|----------|---------|
| 10.5.5.5/32 | B 170 | 100 | | Unusable | 100 I |
| 10.12.1.0/24 | B 170 | 100 | | Unusable | 100 I |
| 10.12.80.4/30 | B 170 | 100 | | Unusable | I |

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

```
Restart Complete
```

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

```
Restart Complete
```

```
bgp.13vpn.0: 3 destinations, 3 routes (0 active, 0 holddown, 3 hidden)
```

```
Restart Complete
```

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

| A Destination | P Prf | Metric 1 | Metric 2 | Next hop | AS path |
|--------------------------|-------|----------|----------|----------|---------|
| 10.4.4.4:4:10.5.5.5/32 | B 170 | 100 | | Unusable | 100 I |
| 10.4.4.4:4:10.12.1.0/24 | B 170 | 100 | | Unusable | 100 I |
| 10.4.4.4:4:10.12.80.4/30 | B 170 | 100 | | Unusable | I |

```
inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
```

```
Restart Complete
```

```
private1___.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

show route summary

| | |
|------------------------------------|--|
| Syntax | show route summary <logical-system (all <i>logical-system-name</i>)> <table <i>routing-table-name</i> > |
| Syntax (EX Series Switches) | show route summary |
| Release Information | Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.0 for EX Series switches. |
| Description | <p>Display summary statistics about the entries in the routing table.</p> <p>CPU utilization might increase while the device learns routes. We recommend that you use the show route summary command after the device learns and enters the routes into the routing table. Depending on the size of your network, this might take several minutes. If you receive a “timeout communicating with routing daemon” error when using the show route summary command, wait several minutes before attempting to use the command again. This is not a critical system error, but you might experience a delay in using the command-line interface (CLI).</p> |
| Options | <p>none—Display summary statistics about the entries in the routing table.</p> <p>logical-system (all <i>logical-system-name</i>)—(Optional) Perform this operation on all logical systems or on a particular logical system.</p> <p>table <i>routing-table-name</i>—(Optional) Display summary statistics for all routing tables whose name begins with this string (for example, inet.0 and inet6.0 are both displayed when you run the show route summary table inet command). If you only want to display statistics for a specific routing table, make sure to enter the exact name of that routing table.</p> |
| Required Privilege Level | view |
| List of Sample Output | show route summary on page 153 show route summary table on page 153 show route summary table (with Route Limits Configured for the Routing Table) on page 154 |
| Output Fields | Table 14 on page 151 lists the output fields for the show route summary command. Output fields are listed in the approximate order in which they appear. |

Table 14: show route summary Output Fields

| Field Name | Field Description |
|----------------------------------|--|
| Router ID | Address of the local routing device. |
| <i>routing-table-name</i> | Name of the routing table (for example, inet.0). |

Table 14: show route summary Output Fields (*continued*)

| Field Name | Field Description |
|-----------------------------|--|
| destinations | Number of destinations for which there are routes in the routing table. |
| routes | Number of routes in the routing table: <ul style="list-style-type: none">• active—Number of routes that are active.• holddown—Number of routes that are in the hold-down state before being declared inactive.• hidden—Number of routes that are not used because of routing policy. |
| Limit/Threshold | Displays the configured route limits for the routing table set with the maximum-prefixes and the maximum-paths statements. If you do not configure route limits for the routing table, the show output does not display this information. <ul style="list-style-type: none">• destinations—The first number represents the maximum number of route prefixes installed in the routing table. The second number represents the number of route prefixes that trigger a warning message.• routes—The first number represents the maximum number of routes. The second number represents the number of routes that trigger a warning message. |
| Direct | Routes on the directly connected network. |
| Local | Local routes. |
| <i>protocol-name</i> | Name of the protocol from which the route was learned. For example, OSPF , RSVP , and Static . |

Sample Output

```

show route summary  user@host> show route summary
Autonomous system number: 69
Router ID: 10.255.71.52
Maximum-ECMP: 32
inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
    Direct:      6 routes,      5 active
    Local:      4 routes,      4 active
    OSPF:       5 routes,      4 active
    Static:     7 routes,      7 active
    IGMP:       1 routes,      1 active
    PIM:        2 routes,      2 active

inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
Restart Complete
    RSVP:       2 routes,      2 active

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

mpls.0: 7 destinations, 7 routes (5 active, 0 holddown, 2 hidden)
Restart Complete
    MPLS:       3 routes,      3 active
    VPLS:       4 routes,      2 active

inet6.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
Restart Complete
    Direct:      2 routes,      2 active
    PIM:         2 routes,      2 active
    MLD:         1 routes,      1 active

green.l2vpn.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
Restart Complete
    BGP:        2 routes,      2 active
    L2VPN:      2 routes,      2 active

red.l2vpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
Restart Complete
    BGP:        2 routes,      2 active
    L2VPN:      1 routes,      1 active

bgp.l2vpn.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
Restart Complete
    BGP:        4 routes,      4 active

```

```

show route summary  user@host> show route summary table inet
table
Router ID: 192.168.0.1

```

```

inet.0: 32 destinations, 34 routes (31 active, 0 holddown, 1 hidden)
    Direct:      6 routes,      5 active
    Local:      9 routes,      9 active
    OSPF:       3 routes,      1 active
    Static:    13 routes,     13 active
    IGMP:       1 routes,      1 active
    PIM:        2 routes,      2 active

```

```
inet.1: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
      Multicast:      1 routes,      1 active

inet6.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
      Local:      1 routes,      1 active
      PIM:      2 routes,      2 active

inet6.1: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
      Multicast:      1 routes,      1 active
```

**show route summary
table (with Route
Limits Configured for
the Routing Table)**

```
user@host> show route summary table VPN-A.inet.0
Autonomous system number: 100
Router ID: 10.255.182.142

VPN-A.inet.0: 13 destinations, 14 routes (13 active, 0 holddown, 0 hidden)
Limit/Threshold: 2000/200 destinations 20/12 routes
      Direct:      2 routes,      2 active
      Local:      1 routes,      1 active
      OSPF:      4 routes,      3 active
      BGP:      4 routes,      4 active
      IGMP:      1 routes,      1 active
      PIM:      2 routes,      2 active
```


show route table

| | |
|------------------------------------|---|
| Syntax | <pre>show route table <i>routing-table-name</i> <brief detail extensive terse> <logical-system (all <i>logical-system-name</i>)></pre> |
| Syntax (EX Series Switches) | <pre>show route table <i>routing-table-name</i> <brief detail extensive terse></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> |
| Description | Display the route entries in a particular routing table. |
| Options | <p>brief detail extensive terse—(Optional) Display the specified level of output.</p> <p>logical-system (all <i>logical-system-name</i>)—(Optional) Perform this operation on all logical systems or on a particular logical system.</p> <p><i>routing-table-name</i>—Display route entries for all routing tables whose name begins with this string (for example, inet.0 and inet6.0 are both displayed when you run the show route table inet command).</p> |
| Required Privilege Level | view |
| Related Documentation | <ul style="list-style-type: none"> • show route summary on page 151 |
| List of Sample Output | <p>show route table bgp.l2.vpn on page 157</p> <p>show route table bgp.l3vpn.0 on page 157</p> <p>show route table bgp.l3vpn.0 detail on page 157</p> <p>show route table bgp.rtarget.0 (When Proxy BGP Route Target Filtering Is Configured) on page 158</p> <p>show route table inet.0 on page 159</p> <p>show route table inet6.0 on page 159</p> <p>show route table inet6.3 on page 159</p> <p>show route table inetflow detail on page 159</p> <p>show route table l2circuit.0 on page 160</p> <p>show route table mpls on page 160</p> <p>show route table mpls extensive on page 161</p> <p>show route table mpls.0 on page 161</p> <p>show route table mpls.0 (RSVP Route—Transit LSP) on page 161</p> <p>show route table vpls_1 detail on page 162</p> <p>show route table vpn-a on page 162</p> <p>show route table vpn-a.mdt.0 on page 162</p> <p>show route table VPN-A detail on page 163</p> <p>show route table VPN-AB.inet.0 on page 163</p> <p>show route table VPN_blue.mvpn-inet6.0 on page 163</p> <p>show route table VPN-A detail on page 164</p> |

[show route table inetflow detail on page 164](#)

Output Fields For information about output fields, see the output field tables for the [show route](#) command, the [show route detail](#) command, the **show route extensive** command, or the **show route terse** command.

Sample Output

show route table bgp.l2vpn

```
user@host> show route table bgp.l2vpn
bgp.l2vpn.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.24.1:1:4:1/96
    *[BGP/170] 01:08:58, localpref 100, from 192.168.24.1
    AS path: I
    > to 10.0.16.2 via fe-0/0/1.0, label-switched-path am
```

show route table bgp.l3vpn.0

```
user@host> show route table bgp.l3vpn.0
bgp.l3vpn.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.71.15:100:10.255.71.17/32
    *[BGP/170] 00:03:59, MED 1, localpref 100, from
10.255.71.15
    AS path: I
    > via so-2/1/0.0, Push 100020, Push 100011(top)
10.255.71.15:200:10.255.71.18/32
    *[BGP/170] 00:03:59, MED 1, localpref 100, from
10.255.71.15
    AS path: I
    > via so-2/1/0.0, Push 100021, Push 100011(top)
```

show route table bgp.l3vpn.0 detail

```
user@host> show route table bgp.l3vpn.0 detail
bgp.l3vpn.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)

10.255.245.12:1:4.0.0.0/8 (1 entry, 1 announced)
  *BGP Preference: 170/-101
    Route Distinguisher: 10.255.245.12:1
    Source: 10.255.245.12
    Next hop: 192.168.208.66 via fe-0/0/0.0, selected
    Label operation: Push 182449
    Protocol next hop: 10.255.245.12
    Push 182449
    Indirect next hop: 863a630 297
    State: <Active Int Ext>
    Local AS: 35 Peer AS: 35
    Age: 12:19 Metric2: 1
    Task: BGP_35.10.255.245.12+179
    Announcement bits (1): 0-BGP.0.0.0.0+179
    AS path: 30 10458 14203 2914 3356 I (Atomic) Aggregator: 3356 4.68.0.11

    Communities: 2914:420 target:11111:1 origin:56:78
    VPN Label: 182449
    Localpref: 100
    Router ID: 10.255.245.12

10.255.245.12:1:4.17.225.0/24 (1 entry, 1 announced)
  *BGP Preference: 170/-101
    Route Distinguisher: 10.255.245.12:1
    Source: 10.255.245.12
    Next hop: 192.168.208.66 via fe-0/0/0.0, selected
    Label operation: Push 182465
    Protocol next hop: 10.255.245.12
    Push 182465
```

```

Indirect next hop: 863a8f0 305
State: <Active Int Ext>
Local AS: 35 Peer AS: 35
Age: 12:19 Metric2: 1
Task: BGP_35.10.255.245.12+179
Announcement bits (1): 0-BGP.0.0.0.0+179
AS path: 30 10458 14203 2914 11853 11853 11853 6496 6496 6496 6496 6496 I
Communities: 2914:410 target:12:34 target:11111:1 origin:12:34
VPN Label: 182465
Localpref: 100
Router ID: 10.255.245.12

10.255.245.12:1:4.17.226.0/23 (1 entry, 1 announced)
*BGP Preference: 170/-101
Route Distinguisher: 10.255.245.12:1
Source: 10.255.245.12
Next hop: 192.168.208.66 via fe-0/0/0.0, selected
Label operation: Push 182465
Protocol next hop: 10.255.245.12
Push 182465
Indirect next hop: 86bd210 330
State: <Active Int Ext>
Local AS: 35 Peer AS: 35
Age: 12:19 Metric2: 1
Task: BGP_35.10.255.245.12+179
Announcement bits (1): 0-BGP.0.0.0.0+179
AS path: 30 10458 14203 2914 11853 11853 11853 6496 6496 6496 6496 6496
6496 I
Communities: 2914:410 target:12:34 target:11111:1 origin:12:34
VPN Label: 182465
Localpref: 100
Router ID: 10.255.245.12

10.255.245.12:1:4.17.251.0/24 (1 entry, 1 announced)
*BGP Preference: 170/-101
Route Distinguisher: 10.255.245.12:1
Source: 10.255.245.12
Next hop: 192.168.208.66 via fe-0/0/0.0, selected
Label operation: Push 182465
Protocol next hop: 10.255.245.12
Push 182465
Indirect next hop: 86bd210 330
State: <Active Int Ext>
Local AS: 35 Peer AS: 35
Age: 12:19 Metric2: 1
Task: BGP_35.10.255.245.12+179
Announcement bits (1): 0-BGP.0.0.0.0+179
AS path: 30 10458 14203 2914 11853 11853 11853 6496 6496 6496 6496 6496
6496 I
Communities: 2914:410 target:12:34 target:11111:1 origin:12:34
VPN Label: 182465
Localpref: 100

```

**show route table
bgp.rtarget.0 (When
Proxy BGP Route**

```

user@host> show route table bgp.rtarget.0
bgp.rtarget.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

```

Target Filtering Is Configured)

```
100:100:100/96
*[RTarget/5] 00:03:14
  Type Proxy
    for 10.255.165.103
    for 10.255.166.124
  Local
```

show route table inet.0

```
user@host> show route table inet.0
inet.0: 12 destinations, 12 routes (11 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[Static/5] 00:51:57
                   > to 111.222.5.254 via fxp0.0
1.0.0.1/32         *[Direct/0] 00:51:58
                   > via at-5/3/0.0
1.0.0.2/32         *[Local/0] 00:51:58
                   Local
12.12.12.21/32     *[Local/0] 00:51:57
                   Reject
13.13.13.13/32     *[Direct/0] 00:51:58
                   > via t3-5/2/1.0
13.13.13.14/32     *[Local/0] 00:51:58
                   Local
13.13.13.21/32     *[Local/0] 00:51:58
                   Local
13.13.13.22/32     *[Direct/0] 00:33:59
                   > via t3-5/2/0.0
127.0.0.1/32      [Direct/0] 00:51:58
                   > via lo0.0
111.222.5.0/24     *[Direct/0] 00:51:58
                   > via fxp0.0
111.222.5.81/32   *[Local/0] 00:51:58
                   Local
```

show route table inet6.0

```
user@host> show route table inet6.0
inet6.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Route, * = Both

fec0:0:0:3::/64   *[Direct/0] 00:01:34
>via fe-0/1/0.0

fec0:0:0:3::/128  *[Local/0] 00:01:34
>Local

fec0:0:0:4::/64   *[Static/5] 00:01:34
>to fec0:0:0:3::ffff via fe-0/1/0.0
```

show route table inet6.3

```
user@router> show route table inet6.3
inet6.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

::10.255.245.195/128
                   *[LDP/9] 00:00:22, metric 1
                   > via so-1/0/0.0
::10.255.245.196/128
                   *[LDP/9] 00:00:08, metric 1
                   > via so-1/0/0.0, Push 100008
```

show route table inetflow detail

```

user@host> show route table inetflow detail
inetflow.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
10.12.44.1,*/48 (1 entry, 1 announced)
    *BGP    Preference: 170/-101
            Next-hop reference count: 2
            State: **Active Ext>
            Local AS: 65002 Peer AS: 65000
            Age: 4
            Task: BGP_65000.10.12.99.5+3792
            Announcement bits (1): 0-Flow
            AS path: 65000 I
            Communities: traffic-rate:0:0
            Validation state: Accept, Originator: 10.12.99.5
            Via: 10.12.44.0/24, Active
            Localpref: 100
            Router ID: 10.255.71.161

10.12.56.1,*/48 (1 entry, 1 announced)
    *Flow    Preference: 5
            Next-hop reference count: 2
            State: **Active>
            Local AS: 65002
            Age: 6:30
            Task: RT Flow
            Announcement bits (2): 0-Flow 1-BGP.0.0.0.0+179
            AS path: I
            Communities: 1:1

```

show route table l2circuit.0

```

user@host> show route table l2circuit.0
l2circuit.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.1.1.195:NoCtrlWord:1:1:Local/96
    * [L2CKT/7] 00:50:47
    > via so-0/1/2.0, Push 100049
    via so-0/1/3.0, Push 100049
10.1.1.195:NoCtrlWord:1:1:Remote/96
    * [LDP/9] 00:50:14
    Discard
10.1.1.195:CtrlWord:1:2:Local/96
    * [L2CKT/7] 00:50:47
    > via so-0/1/2.0, Push 100049
    via so-0/1/3.0, Push 100049
10.1.1.195:CtrlWord:1:2:Remote/96
    * [LDP/9] 00:50:14
    Discard

```

show route table mpls

```

user@host> show route table mpls
mpls.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0          * [MPLS/0] 00:13:55, metric 1
           Receive
1          * [MPLS/0] 00:13:55, metric 1
           Receive
2          * [MPLS/0] 00:13:55, metric 1
           Receive
1024       * [VPN/0] 00:04:18
           to table red.inet.0, Pop

```

show route table mpls extensive

```

user@host> show route table mpls extensive
100000 (1 entry, 1 announced)
TSI:
KRT in-kernel 100000 /36 -> {so-1/0/0.0}
    *LDP      Preference: 9
              Next hop: via so-1/0/0.0, selected
              Pop
              State: <Active Int>
              Age: 29:50      Metric: 1
              Task: LDP
              Announcement bits (1): 0-KRT
              AS path: I
              Prefixes bound to route: 10.0.0.194/32

```

show route table mpls.0

```

user@host> show route table mpls.0
mpls.0: 11 destinations, 11 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0          *[MPLS/0] 00:45:09, metric 1
           Receive
1          *[MPLS/0] 00:45:09, metric 1
           Receive
2          *[MPLS/0] 00:45:09, metric 1
           Receive
100000     *[L2VPN/7] 00:43:04
           > via so-0/1/0.1, Pop
100001     *[L2VPN/7] 00:43:03
           > via so-0/1/0.2, Pop      Offset: 4
100002     *[LDP/9] 00:43:22, metric 1
           via so-0/1/2.0, Pop
           > via so-0/1/3.0, Pop
100002(S=0) *[LDP/9] 00:43:22, metric 1
           via so-0/1/2.0, Pop
           > via so-0/1/3.0, Pop
100003     *[LDP/9] 00:43:22, metric 1
           > via so-0/1/2.0, Swap 100002
           via so-0/1/3.0, Swap 100002
100004     *[LDP/9] 00:43:16, metric 1
           via so-0/1/2.0, Swap 100049
           > via so-0/1/3.0, Swap 100049
so-0/1/0.1 *[L2VPN/7] 00:43:04
           > via so-0/1/2.0, Push 100001, Push 100049(top)
           via so-0/1/3.0, Push 100001, Push 100049(top)
so-0/1/0.2 *[L2VPN/7] 00:43:03
           via so-0/1/2.0, Push 100000, Push 100049(top) Offset: -4
           > via so-0/1/3.0, Push 100000, Push 100049(top) Offset: -4

```

show route table mpls.0 (RSVP Route—Transit LSP)

```

user@host> show route table mpls.0
mpls.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0          *[MPLS/0] 00:37:31, metric 1
           Receive
1          *[MPLS/0] 00:37:31, metric 1
           Receive
2          *[MPLS/0] 00:37:31, metric 1
           Receive

```

```

13          *[MPLS/0] 00:37:31, metric 1
            Receive
300352      *[RSVP/7/1] 00:08:00, metric 1
            > to 8.64.0.106 via ge-1/0/1.0, label-switched-path lsp1_p2p
300352(S=0) *[RSVP/7/1] 00:08:00, metric 1
            > to 8.64.0.106 via ge-1/0/1.0, label-switched-path lsp1_p2p
300384      *[RSVP/7/2] 00:05:20, metric 1
            > to 8.64.1.106 via ge-1/0/0.0, Pop
300384(S=0) *[RSVP/7/2] 00:05:20, metric 1
            > to 8.64.1.106 via ge-1/0/0.0, Pop

```

show route table vpls_1 detail

```

user@host> show route table vpls_1 detail
vpls_1.l2vpn.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
Restart Complete

1.1.1.11:1000:1:1/96 (1 entry, 1 announced)
*L2VPN Preference: 170/-1
Receive table: vpls_1.l2vpn.0
Next-hop reference count: 2
State: <Active Int Ext>
Age: 4:29:47 Metric2: 1
Task: vpls_1-l2vpn
Announcement bits (1): 1-BGP.0.0.0.0+179
AS path: I
Communities: Layer2-info: encaps:VPLS, control flags:Site-Down
Label-base: 800000, range: 8, status-vector: 0xFF

```

show route table vpn-a

```

user@host> show route table vpn-a
vpn-a.l2vpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)

+ = Active Route, - = Last Active, * = Both
192.168.16.1:1:1:1/96
    *[VPN/7] 05:48:27
    Discard
192.168.24.1:1:2:1/96
    *[BGP/170] 00:02:53, localpref 100, from 192.168.24.1
    AS path: I
    > to 10.0.16.2 via fe-0/0/1.0, label-switched-path am
192.168.24.1:1:3:1/96
    *[BGP/170] 00:02:53, localpref 100, from 192.168.24.1
    AS path: I
    > to 10.0.16.2 via fe-0/0/1.0, label-switched-path am

```

show route table vpn-a.mdt.0

```

user@host> show route table vpn-a.mdt.0
vpn-a.mdt.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1:1:0:10.255.14.216:232.1.1.1/144
    *[MVPN/70] 01:23:05, metric2 1
    Indirect
1:1:1:10.255.14.218:232.1.1.1/144
    *[BGP/170] 00:57:49, localpref 100, from 10.255.14.218
    AS path: I
    > via so-0/0/0.0, label-switched-path r0e-to-r1
1:1:2:10.255.14.217:232.1.1.1/144
    *[BGP/170] 00:57:49, localpref 100, from 10.255.14.217
    AS path: I
    > via so-0/0/1.0, label-switched-path r0-to-r2

```


show route table VPN-A detail

```

user@host> show route table VPN-A detail
VPN-AB.inet.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
10.255.179.9/32 (1 entry, 1 announced)
    *BGP      Preference: 170/-101
                Route Distinguisher: 10.255.179.13:200
                Next hop type: Indirect
                Next-hop reference count: 5
                Source: 10.255.179.13
                Next hop type: Router, Next hop index: 732
                Next hop: 10.39.1.14 via fe-0/3/0.0, selected
                Label operation: Push 299824, Push 299824(top)
                Protocol next hop: 10.255.179.13
                Push 299824
                Indirect next hop: 8f275a0 1048574
                State: (Secondary Active Int Ext)
                Local AS: 1 Peer AS: 1
                Age: 3:41:06 Metric: 1 Metric2: 1
                Task: BGP_1.10.255.179.13+64309
                Announcement bits (2): 0-KRT 1-BGP RT Background
                AS path: I
                Communities: target:1:200 rte-type:0.0.0.0:1:0
                Import Accepted
                VPN Label: 299824 TTL Action: vrf-ttl-propagate
                Localpref: 100
                Router ID: 10.255.179.13
                Primary Routing Table bgp.l3vpn.0

```

show route table VPN-AB.inet.0

```

user@host> show route table VPN-AB.inet.0
VPN-AB.inet.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.39.1.0/30      *[OSPF/10] 00:07:24, metric 1
                  > via so-7/3/1.0
10.39.1.4/30      *[Direct/0] 00:08:42
                  > via so-5/1/0.0
10.39.1.6/32      *[Local/0] 00:08:46
                  Local
10.255.71.16/32   *[Static/5] 00:07:24
                  > via so-2/0/0.0
10.255.71.17/32   *[BGP/170] 00:07:24, MED 1, localpref 100, from
10.255.71.15
                  AS path: I
                  > via so-2/1/0.0, Push 100020, Push 100011(top)
10.255.71.18/32   *[BGP/170] 00:07:24, MED 1, localpref 100, from
10.255.71.15
                  AS path: I
                  > via so-2/1/0.0, Push 100021, Push 100011(top)
10.255.245.245/32 *[BGP/170] 00:08:35, localpref 100
                  AS path: 2 I
                  > to 10.39.1.5 via so-5/1/0.0
10.255.245.246/32 *[OSPF/10] 00:07:24, metric 1
                  > via so-7/3/1.0

```

show route table VPN_blue.mvpn-inet6.0

```

user@host> show route table VPN_blue.mvpn-inet6.0
vpn_blue.mvpn-inet6.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1:10.255.2.202:65535:10.255.2.202/432

```

```

* [BGP/170] 00:02:37, localpref 100, from 10.255.2.202
  AS path: I
  > via so-0/1/3.0
1:10.255.2.203:65535:10.255.2.203/432
* [BGP/170] 00:02:37, localpref 100, from 10.255.2.203
  AS path: I
  > via so-0/1/0.0
1:10.255.2.204:65535:10.255.2.204/432
* [MVPN/70] 00:57:23, metric2 1
  Indirect
5:10.255.2.202:65535:128::192.168.90.2:128:ffff::1/432
* [BGP/170] 00:02:37, localpref 100, from 10.255.2.202
  AS path: I
  > via so-0/1/3.0
6:10.255.2.203:65535:65000:128::10.12.53.12:128:ffff::1/432
* [PIM/105] 00:02:37
  Multicast (IPv6)
7:10.255.2.202:65535:65000:128::192.168.90.2:128:ffff::1/432
* [MVPN/70] 00:02:37, metric2 1
  Indirect

```

show route table VPN-A detail

```

user@host> show route table VPN-A detail
VPN-AB.inet.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
10.255.179.9/32 (1 entry, 1 announced)
  *BGP
    Preference: 170/-101
    Route Distinguisher: 10.255.179.13:200
    Next hop type: Indirect
    Next-hop reference count: 5
    Source: 10.255.179.13
    Next hop type: Router, Next hop index: 732
    Next hop: 10.39.1.14 via fe-0/3/0.0, selected
    Label operation: Push 299824, Push 299824(top)
    Protocol next hop: 10.255.179.13
    Push 299824
    Indirect next hop: 8f275a0 1048574
    State: (Secondary Active Int Ext)
    Local AS: 1 Peer AS: 1
    Age: 3:41:06 Metric: 1 Metric2: 1
    Task: BGP_1.10.255.179.13+64309
    Announcement bits (2): 0-KRT 1-BGP RT Background
    AS path: I
    Communities: target:1:200 rte-type:0.0.0.0:1:0
    Import Accepted
    VPN Label: 299824 TTL Action: vrf-ttl-propagate
    Localpref: 100
    Router ID: 10.255.179.13
    Primary Routing Table bgp.13vpn.0

```

show route table inetflow detail

```

user@host> show route table inetflow detail
inetflow.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
10.12.44.1,*/48 (1 entry, 1 announced)
  *BGP
    Preference: 170/-101
    Next-hop reference count: 2
    State: **Active Ext>
    Local AS: 65002 Peer AS: 65000
    Age: 4
    Task: BGP_65000.10.12.99.5+3792
    Announcement bits (1): 0-Flow
    AS path: 65000 I
    Communities: traffic-rate:0:0

```

```

Validation state: Accept, Originator: 10.12.99.5
Via: 10.12.44.0/24, Active
Localpref: 100
Router ID: 10.255.71.161

10.12.56.1,*/48 (1 entry, 1 announced)
  *Flow Preference: 5
    Next-hop reference count: 2
    State: **Active>
    Local AS: 65002
    Age: 6:30
    Task: RT Flow
    Announcement bits (2): 0-Flow 1-BGP.0.0.0.0+179
    AS path: I
    Communities: 1:1

user@PE1> show route table green.l2vpn.0 (VPLS Multihoming with FEC 129)
green.l2vpn.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1.1.1.2:100:1.1.1.2/96 AD
    *[VPLS/170] 1d 03:11:03, metric2 1
    Indirect
1.1.1.4:100:1.1.1.4/96 AD
    *[BGP/170] 1d 03:11:02, localpref 100, from 1.1.1.4
    AS path: I, validation-state: unverified
    > via ge-1/2/1.5
1.1.1.2:100:1:0/96 MH
    *[VPLS/170] 1d 03:11:03, metric2 1
    Indirect
1.1.1.4:100:1:0/96 MH
    *[BGP/170] 1d 03:11:02, localpref 100, from 1.1.1.4
    AS path: I, validation-state: unverified
    > via ge-1/2/1.5
1.1.1.4:NoCtrlWord:5:100:100:1.1.1.2:1.1.1.4/176
    *[VPLS/7] 1d 03:11:02, metric2 1
    > via ge-1/2/1.5
1.1.1.4:NoCtrlWord:5:100:100:1.1.1.4:1.1.1.2/176
    *[LDP/9] 1d 03:11:02
    Discard

```

traceroute

| | |
|----------------------------|---|
| Syntax | <pre>traceroute <i>host</i> <as-number-lookup> <bypass-routing> <clns> <gateway <i>address</i>> <inet inet6> <interface <i>interface-name</i>> <logical system <i>logical-system-name</i>> <monitor <i>host</i>> <mpls (ldp <i>FEC address</i> rsvp <i>label-switched-path-name</i>)> <no-resolve> <propagate-ttl> <routing-instance <i>routing-instance-name</i>> <source <i>source-address</i>> <tos <i>value</i>> <ttl <i>value</i>> <wait <i>seconds</i>></pre> |
| Syntax (QFX Series) | <pre>traceroute <i>host</i> <as-number-lookup> <bypass-routing> <gateway <i>address</i>> <inet> <interface <i>interface-name</i>> <monitor <i>host</i>> <no-resolve> <routing-instance <i>routing-instance-name</i>> <source <i>source-address</i>> <tos <i>value</i>> <ttl <i>value</i>> <wait <i>seconds</i>></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>mpls option introduced in Junos OS Release 9.2.</p> <p>Command introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>propagate-ttl option introduced in Junos OS Release 12.1.</p> |
| Description | Display the route that packets take to a specified network host. Use traceroute as a debugging tool to locate points of failure in a network. |
| Options | <p>host—IP address or name of remote host.</p> <p>as-number-lookup—(Optional) Display the autonomous system (AS) number of each intermediate hop on the path from the host to the destination.</p> <p>bypass-routing—(Optional) Bypass the normal routing tables and send requests directly to a system on an attached network. If the system is not on a directly attached network, an error is returned. Use this option to display a route to a local system through an interface that has no route through it.</p> |

clns—(Optional) Trace the route belonging to the Connectionless Network Service (CLNS).

gateway address—(Optional) Address of a router or switch through which the route transits.

inet | inet6—(Optional) Trace the route belonging to IPv4 or IPv6, respectively.

interface *interface-name*—(Optional) Name of the interface over which to send packets.

logical-system *logical-system-name*—(Optional) Perform this operation on all logical systems or on a particular logical system.

monitor *host*—(Optional) Display real-time monitoring information for the specified host.

mpls (*ldp FEC address | rsvp label-switched-path name*)—(Optional) See **traceroute mpls ldp** and **traceroute mpls rsvp**.

no-resolve—(Optional) Do not attempt to determine the hostname that corresponds to the IP address.

propagate-ttl—(Optional) On the PE router, use this option to view locally generated Routing Engine transit traffic. This is applicable for MPLS L3VPN traffic only. Use for troubleshooting, when you want to view hop-by-hop information from the local provider router to the remote provider router, when TTL decrementing is disabled on the core network using the **no-propagate-ttl** configuration statement.



NOTE: Using **propagate-ttl** with **traceroute** on the CE router does not show hop-by-hop information.

routing-instance *routing-instance-name*—(Optional) Name of the routing instance for the traceroute attempt.

source *source-address*—(Optional) Source address of the outgoing traceroute packets.

tos *value*—(Optional) Value to include in the IP type-of-service (ToS) field. The range of values is 0 through 255.

ttl *value*—(Optional) Maximum time-to-live value to include in the traceroute request. The range of values is 0 through 128.

wait *seconds*—(Optional) Maximum time to wait for a response to the traceroute request.

Required Privilege Level network

Related Documentation • [traceroute monitor](#)

List of Sample Output [traceroute on page 169](#)
[traceroute as-number-lookup host on page 169](#)

[traceroute no-resolve on page 169](#)

[traceroute propagate-ttl on page 169](#)

[traceroute \(Between CE Routers, Layer 3 VPN\) on page 169](#)

[traceroute \(Through an MPLS LSP\) on page 169](#)

Output Fields [Table 15 on page 168](#) describes the output fields for the **traceroute** command. Output fields are listed in the approximate order in which they appear.

Table 15: traceroute Output Fields

| Field Name | Field Description |
|------------------------|---|
| traceroute to | IP address of the receiver. |
| hops max | Maximum number of hops allowed. |
| byte packets | Size of packets being sent. |
| <i>number-of-hops</i> | Number of hops from the source to the named router or switch. |
| <i>router-name</i> | Name of the router or switch for this hop. |
| <i>address</i> | Address of the router or switch for this hop. |
| Round trip time | Average round-trip time, in milliseconds (ms). |

Sample Output

traceroute

```
user@host> traceroute santacruz
traceroute to green.company.net (10.156.169.254), 30 hops max, 40 byte packets
 1 blue23 (10.168.1.254) 2.370 ms 2.853 ms 0.367 ms
 2 red14 (10.168.255.250) 0.778 ms 2.937 ms 0.446 ms
 3 yellow (10.156.169.254) 7.737 ms 89.905 ms 0.834 ms
```

traceroute as-number-lookup host

```
user@host> traceroute as-number-lookup 10.100.1.1
traceroute to 10.100.1.1 (10.100.1.1), 30 hops max, 40 byte packets
 1 10.39.1.1 (10.39.1.1) 0.779 ms 0.728 ms 0.562 ms
 2 10.39.1.6 (10.39.1.6) [AS 32] 0.657 ms 0.611 ms 0.617 ms
 3 10.100.1.1 (10.100.1.1) [AS 10, 40, 50] 0.880 ms 0.808 ms 0.774 ms
```

traceroute no-resolve

```
user@host> traceroute santacruz no-resolve
traceroute to green.company.net (10.156.169.254), 30 hops max, 40 byte packets
 1 10.168.1.254 0.458 ms 0.370 ms 0.365 ms
 2 10.168.255.250 0.474 ms 0.450 ms 0.444 ms
 3 10.156.169.254 0.931 ms 0.876 ms 0.862 ms
```

traceroute propagate-ttl

```
user@host> traceroute propagate-ttl 100.200.2.2 routing-instance VPN-A
traceroute to 100.200.2.2 (100.200.2.2) from 1.1.0.2, 30 hops max, 40 byte packets

 1 1.2.0.2 (1.2.0.2) 2.456 ms 1.753 ms 1.672 ms
   MPLS Label=299776 CoS=0 TTL=1 S=0
   MPLS Label=299792 CoS=0 TTL=1 S=1
 2 1.3.0.2 (1.3.0.2) 1.213 ms 1.225 ms 1.166 ms
   MPLS Label=299792 CoS=0 TTL=1 S=1
 3 100.200.2.2 (100.200.2.2) 1.422 ms 1.521 ms 1.443 ms
```

traceroute (Between CE Routers, Layer 3 VPN)

```
user@host> traceroute vpn09
traceroute to vpn09.skybank.net (10.255.14.179), 30 hops max, 40
byte packets
 1 10.39.10.21 (10.39.10.21) 0.598 ms 0.500 ms 0.461 ms
 2 10.39.1.13 (10.39.1.13) 0.796 ms 0.775 ms 0.806 ms
   MPLS Label=100006 CoS=0 TTL=1 S=1
 3 vpn09.skybank.net (10.255.14.179) 0.783 ms 0.716 ms 0.686
```

traceroute (Through an MPLS LSP)

```
user@host> traceroute mpls1
traceroute to 10.168.1.224 (10.168.1.224), 30 hops max, 40 byte packets
 1 mpls1-sr0.company.net (10.168.200.101) 0.555 ms 0.393 ms 0.367 ms
   MPLS Label=1024 CoS=0 TTL=1
 2 mpls5-lo0.company.net (10.168.1.224) 0.420 ms 0.394 ms 0.401 ms
```


PART 4

Troubleshooting

- [Routing Protocol Process Memory FAQs on page 173](#)

CHAPTER 6

Routing Protocol Process Memory FAQs

- [Routing Protocol Process Memory FAQs Overview on page 173](#)
- [Routing Protocol Process Memory FAQs on page 174](#)

Routing Protocol Process Memory FAQs Overview

Junos OS is based on the FreeBSD Unix operating system. The open source software is modified and hardened to operate in the device's specialized environment. For example, some executables have been deleted, while other utilities were de-emphasized. Additionally, certain software processes were added to enhance the routing functionality. The result of this transformation is the kernel, the heart of the Junos OS software.

The kernel is responsible for operating multiple processes that perform the actual functions of the device. Each process operates in its own protected memory space, while the communication among all the processes is still controlled by the kernel. This separation provides isolation between the processes, and resiliency in the event of a process failure. This is important in a core routing platform because a single process failure does not cause the entire device to cease functioning.

Some of the common software processes include the routing protocol process (rpd) that controls the device's protocols, the device control process (dcd) that controls the device's interfaces, the management process (mgd) that controls user access to the device, the chassis process (chassisd) that controls the device's properties itself, and the Packet Forwarding Engine process (pfed) that controls the communication between the device's Packet Forwarding Engine and the Routing Engine. The kernel also generates specialized processes as needed for additional functionality, such as SNMP, the Virtual Router Redundancy Protocol (VRRP), and Class of Service (CoS).

The routing protocol process is a software process within the Routing Engine software, which controls the routing protocols that run on the device. Its functionality includes all protocol messages, routing table updates, and implementation of routing policies.

The routing protocol process starts all configured routing protocols and handles all routing messages. It maintains one or more routing tables, which consolidate the routing information learned from all routing protocols. From this routing information, the routing protocol process determines the active routes to network destinations and installs these routes into the Routing Engine's forwarding table. Finally, it implements routing policy, which allows you to control the routing information that is transferred between the routing

protocols and the routing table. Using routing policy, you can filter and limit the transfer of information as well as set properties associated with specific routes.

Related Documentation

- [Routing Protocol Process Memory FAQs on page 174](#)

Routing Protocol Process Memory FAQs

The following sections present the most frequently asked questions and answers related to the routing protocol process memory utilization, operation, interpretation of related command outputs, and troubleshooting the software process.

Frequently Asked Questions: Routing Protocol Process Memory

This section presents frequently asked questions and answers related to the memory usage of the routing protocol process.

Why does the routing protocol process use excessive memory?

The routing protocol process uses hundreds of megabytes of RAM in the Routing Engine to store information needed for the operation of routing and related protocols, such as BGP, OSPF, IS-IS, RSVP, LDP and MPLS. Such huge consumption of memory is common for the process, as the information it stores includes routes, next hops, interfaces, routing policies, labels, and label-switched paths (LSPs). Because access to the RAM memory is much faster than access to the hard disk, most of the routing protocol process information is stored in the RAM memory instead of using the hard disk space. This ensures that the performance of the routing protocol process is maximized.

How can I check the amount of memory the routing protocol process is using?

You can check routing protocol process memory usage by entering the **show system processes** and the **show task memory** Junos OS command-line interface (CLI) operational mode commands.

The **show system processes** command displays information about software processes that are running on the device and that have controlling terminals. The **show task memory** command displays memory utilization for routing protocol tasks on the Routing Engine.

You can check the routing protocol process memory usage by using the **show system processes** command with the **extensive** option. The **show task memory** command displays a report generated by the routing protocol process on its own memory usage. However, this report does not display all the memory used by the process. The value reported by the routing protocol process does not account for the memory used for the **TEXT** and **STACK** segments, or the memory used by the process's internal memory manager. Further, the Resident Set Size value includes shared library pages used by the routing protocol process.

For more information about checking the routing protocol process memory usage, see [Check Routing Protocol Process \(rpd\) Memory Usage](#).

For more information, see the **show system processes** command and the **show task memory** command.

I just deleted a large number of routes from the routing protocol process. Why is it still using so much memory?

The **show system processes extensive** command displays a **RES** value measured in kilobytes. This value represents the amount of program memory resident in the physical memory. This is also known as RSS or Resident Set Size. The **RES** value includes shared library pages used by the process. Any amount of memory freed by the process might still be considered part of the **RES** value. Generally, the kernel delays the migrating of memory out of the **Inact** queue into the **Cache** or **Free** list unless there is a memory shortage. This can lead to large discrepancies between the values reported by the routing protocol process and the kernel, even after the routing protocol process has freed a large amount of memory.

Frequently Asked Questions: Interpreting Routing Protocol Process-Related Command Outputs

This section presents frequently asked questions and answers about the routing protocol process-related Junos OS command-line interface (CLI) command outputs that are used to display the memory usage of the routing protocol process.

How do I interpret memory numbers displayed in the show system processes extensive command output?

The **show system processes extensive** command displays exhaustive system process information about software processes that are running on the device and have controlling terminals. This command is equivalent to the UNIX **top** command. However, the UNIX **top** command shows real-time memory usage, with the memory values constantly changing, while the **show system processes extensive** command provides a snapshot of memory usage in a given moment.

To check overall CPU and memory usage, enter the **show system processes extensive** command. Refer to [Table 16 on page 176](#) for information about the **show system processes extensive** commands output fields.

```
user@host> show system processes extensive
last pid: 544; load averages: 0.00, 0.00, 0.00 18:30:33
37 processes: 1 running, 36 sleeping

Mem: 25M Active, 3968K Inact, 19M Wired, 184K Cache, 8346K Buf, 202M Free
Swap: 528M Total, 64K Used, 528M Free

  PID USERNAME PRI NICE SIZE  RES STATE   TIME  WCPU   CPU COMMAND
  544 root      30  0   604K 768K RUN      0:00 0.00% 0.00% top
    3 root      28  0    0K   12K psleep   0:00 0.00% 0.00% vmdaemon
    4 root      28  0    0K   12K update  0:03 0.00% 0.00% update
  528 aviva     18  0   660K 948K pause    0:00 0.00% 0.00% tcsh
  204 root      18  0   300K 544K pause    0:00 0.00% 0.00% csh
  131 root      18  0   332K 532K pause    0:00 0.00% 0.00% cron
  186 root      18  0   196K  68K pause    0:00 0.00% 0.00% watchdog
    27 root      10  0   512M 16288K mfsidl   0:00 0.00% 0.00% mount_mfs
    1 root      10  0   620K 344K wait    0:00 0.00% 0.00% init
  304 root       3  0   884K 900K ttyin   0:00 0.00% 0.00% bash
  200 root       3  0   180K 540K ttyin   0:00 0.00% 0.00% getty
  203 root       3  0   180K 540K ttyin   0:00 0.00% 0.00% getty
  202 root       3  0   180K 540K ttyin   0:00 0.00% 0.00% getty
  201 root       3  0   180K 540K ttyin   0:00 0.00% 0.00% getty
  194 root       2  0  2248K 1640K select 0:11 0.00% 0.00% rpd
  205 root       2  0   964K  800K select 0:12 0.00% 0.00% tnp.chassisd
```

```

189 root      2 -12  352K  740K select  0:03  0.00%  0.00% xntpd
114 root      2  0   296K  612K select  0:00  0.00%  0.00% amd
188 root      2  0   780K  600K select  0:00  0.00%  0.00% dcd
527 root      2  0   176K  580K select  0:00  0.00%  0.00% rlogind
195 root      2  0   212K  552K select  0:00  0.00%  0.00% inetd
187 root      2  0   192K  532K select  0:00  0.00%  0.00% tnetd
 83 root      2  0   188K  520K select  0:00  0.00%  0.00% syslogd
538 root      2  0  1324K  516K select  0:00  0.00%  0.00% mgd
 99 daemon    2  0   176K  492K select  0:00  0.00%  0.00% portmap
163 root      2  0   572K  420K select  0:00  0.00%  0.00% nsrexecd
192 root      2  0   560K  400K select  0:10  0.00%  0.00% snmpd
191 root      2  0  1284K  376K select  0:00  0.00%  0.00% mgd
537 aviva     2  0   636K  364K select  0:00  0.00%  0.00% cli
193 root      2  0   312K  204K select  0:07  0.00%  0.00% mib2d
  5 root      2  0      0K   12K pfesel  0:00  0.00%  0.00% if_pfe
  2 root     -18  0      0K   12K psleep  0:00  0.00%  0.00% pagedaemon
  0 root     -18  0      0K    0K sched   0:00  0.00%  0.00% swapper

```

Table 16 on page 176 describes the output fields that represent the memory values for the **show system processes extensive** command. Output fields are listed in the approximate order in which they appear.

Table 16: show system processes extensive Output Fields

| Field Name | Field Description |
|---------------|---|
| Mem | Information about physical and virtual memory allocation. |
| Active | Memory allocated and actively used by the program. |
| Inact | Memory allocated but not recently used or memory freed by the programs. Inactive memory remains mapped in the address space of one or more processes and, therefore, counts toward the RSS value of those processes. |
| Wired | Memory that is not eligible to be swapped, usually used for in-kernel memory structures and/or memory physically locked by a process. |
| Cache | Memory that is not associated with any program and does not need to be swapped before being reused. |
| Buf | Size of memory buffer used to hold data recently called from the disk. |
| Free | Memory that is not associated with any programs. Memory freed by a process can become Inactive , Cache , or Free , depending on the method used by the process to free the memory. |
| Swap | Information about swap memory. <ul style="list-style-type: none"> • Total—Total memory available to be swapped to disk. • Used—Memory swapped to disk. • Free—Memory available for further swap. |

The rest of the command output displays information about the memory usage of each process. The **SIZE** field indicates the size of the virtual address space, and the **RES** field indicates the amount of the program in physical memory, which is also known as RSS or Resident Set Size. For more information, see the **show system processes** command.

What is the difference between Active and Inact memory that is displayed by the show system processes extensive command?

When the system is under memory pressure, the pageout process reuses memory from the free, cache, inactive and, if necessary, active pages. When the pageout process runs, it scans memory to see which pages are good candidates to be unmapped and freed up. Thus, the distinction between **Active** and **Inact** memory is only used by the pageout process to determine which pool of pages to free first at the time of a memory shortage.

The pageout process first scans the **Inact** list, and checks whether the pages on this list have been accessed since the time they have been listed here. The pages that have been accessed are moved from the **Inact** list to the **Active** list. On the other hand, pages that have not been accessed become prime candidates to be freed by the pageout process. If the pageout process cannot produce enough free pages from the **Inact** list, pages from the **Active** list get freed up.

Because the pageout process runs only when the system is under memory pressure, the pages on the **Inact** list remain untouched – even if they have not been accessed recently – when the amount of **Free** memory is adequate.

How do I interpret memory numbers displayed in the show task memory command output?

The **show task memory** command provides a comprehensive picture of the memory utilization for routing protocol tasks on the Routing Engine. The routing protocol process is the main task that uses Routing Engine memory.

To check routing process memory usage, enter the **show task memory** command. Refer to [Table 17 on page 177](#) for information about the **show task memory** command output fields.

```
user@host> show task memory
Memory          Size (kB)  %Available  When
Currently In Use:    29417      3%         now
Maximum Ever Used:   33882      4%         00/02/11 22:07:03
Available:          756281     100%        now
```

[Table 17 on page 177](#) describes the output fields for the **show task memory** command. Output fields are listed in the approximate order in which they appear.

Table 17: show task memory Output Fields

| Field Name | Field Description |
|--------------------------|---|
| Memory Currently In Use | Memory currently in use. Dynamically allocated memory plus the DATA segment memory in kilobytes. |
| Memory Maximum Ever Used | Maximum memory ever used. |
| Memory Available | Memory currently available. |

The **show task memory** command does not display all the memory used by the routing protocol process. This value does not account for the memory used for the **TEXT** and

STACK segments, or the memory used by the routing protocol process's internal memory manager.

Why is the Currently In Use value less than the RES value?

The **show task memory** command displays a **Currently In Use** value measured in kilobytes. This value represents the memory currently in use. It is the dynamically allocated memory plus the **DATA** segment memory. The **show system processes extensive** command displays a **RES** value measured in kilobytes. This value represents the amount of program memory resident in the physical memory. This is also known as RSS or Resident Set Size.

The **Currently In Use** value does not account for all of the memory that the routing protocol process uses. This value does not include the memory used for the **TEXT** and the **STACK** segments, and a small percentage of memory used by the routing protocol process's internal memory manager. Further, the **RES** value includes shared library pages used by the routing protocol process.

Any amount of memory freed by the routing protocol process might still be considered part of the **RES** value. Generally, the kernel delays the migrating of memory out of the **Inact** queue into the **Cache** or **Free** list unless there is a memory shortage. This can lead to large discrepancies between the **Currently In Use** value and the **RES** value.

Frequently Asked Questions: Routing Protocol Process Memory Swapping

This section presents frequently asked questions and answers related to the memory swapping of the routing protocol process from the Routing Engine memory to the hard disk memory.

How do I monitor swap activity?

When the system is under memory pressure, the pageout process reuses memory from the free, cache, inact and, if necessary, active pages. You can monitor the swap activity by viewing the syslog message reported by the kernel during periods of high pageout activity.

The syslog message appears as follows:

```
Mar  3 20:08:02 olympic /kernel: High pageout rate!! 277 pages/sec.
```

You can use the **vmstat -s** command to print the statistics for the swapout activity. The displayed statistics appear as follows:

```
0 swap pager pageouts
0 swap pager pages paged out
```

The **swap pager pageouts** is the number of pageout operations to the swap device, and the **swap pager pages paged out** is the number of pages paged out to the swap device.

Why does the system start swapping when I try to dump core using the request system core-dumps command?

The **request system core-dumps** command displays a list of system core files created when the device has failed. This command can be useful for diagnostic purposes. Each list item includes the file permissions, number of links, owner, group, size, modification

date, path, and filename. You can use the **core-filename** option and the **core-file-info**, **brief**, and **detail** options to display more information about the specified core-dump files.

You can use the **request system core-dumps** command to perform a non-fatal core-dump without aborting the routing protocol process. To do this, the routing protocol process is forked, generating a second copy, and then aborted. This process can double the memory consumed by the two copies of the routing protocol processes, pushing the system into swap.

Why does the show system processes extensive command show that memory is swapped to disk although there is plenty of free memory?

Memory can remain swapped out indefinitely if it is not accessed again. Therefore, the **show system processes extensive** command shows that memory is swapped to disk even though there is plenty of free memory, and such a situation is not unusual.

Frequently Asked Questions: Troubleshooting the Routing Protocol Process

This section presents frequently asked questions and answers related to a shortage of memory and memory leakage by the routing protocol process.

What does the RPD_OS_MEMHIGH message mean?

The **RPD_OS_MEMHIGH** message is written into the system message file if the routing protocol process is running out of memory. This message alerts you that the routing protocol process is using the indicated amount and percentage of Routing Engine memory, which is considered excessive. This message is generated either because the routing protocol process is leaking memory or the use of system resources is excessive, perhaps because routing filters are misconfigured or the configured network topology is very complex.

When the memory utilization for the routing protocol process is using all available Routing Engine DRAM memory (Routing Engines with maximum 2 GB DRAM) or reaches the limit of 2 GB of memory (Routing Engines with 4 GB DRAM), a message of the following form is written every minute in the syslog message file:

RPD_OS_MEMHIGH: Using 188830 KB of memory, 100 percent of available

This message includes the amount, in kilobytes and/or the percentage, of the available memory in use.

This message should not appear under normal conditions, as any further memory allocations usually require a portion of existing memory to be written to swap. As a recommended solution, increase the amount of RAM in the Routing Engine. For more information, go to <http://kb.juniper.net/InfoCenter/index?page=content&id=KB14186>.

What can I do when there is a memory shortage even after a swap?

It is not recommended for the system to operate in this state, notwithstanding the existence of swap. The protocols that run in the routing protocol process usually have a real-time requirement that cannot reliably withstand the latency of being swapped to hard disk. If the memory shortage has not resulted from a memory leak, then either a

reduction in the memory usage or an upgrade to a higher memory-capacity Routing Engine is required.

How do I determine whether there is a memory leak in the routing protocol process?

Memory leaks are typically the result of a seemingly unbounded growth in the memory usage of a process as reported by the **show system processes extensive** command.

There are two classes of memory leaks that the routing protocol process can experience.

- The first class occurs when the allocated memory that is no longer in use is not freed. This class of leak can usually be fixed by taking several samples of the **show task memory detail** command over a period of time and comparing the deltas.
- The second class occurs when there is a late access to freed memory. If the access is not outside the mapped address space, the kernel backfills the accessed page with real memory. This backfill is done without the knowledge of the routing protocol process's internal memory allocator, which makes this class of leak much more difficult to resolve. If a memory leak of this class is suspected, writing the state of the system to a disk file (creating a core file) is suggested.

A large discrepancy between the **RES** value and the **Currently In Use** value might indicate a memory leak. However, large discrepancies can also occur for legitimate reasons. For example, the memory used for the **TEXT** and **STACK** segments or the memory used by the routing protocol process's internal memory manager might not be displayed. Further, the **RES** value includes shared library pages used by the process.

What is the task_timer?

The source of a routing protocol process memory leak can usually be identified by dumping the timers for each task. You can use the **show task task-name** command to display routing protocol tasks on the Routing Engine. Tasks can be baseline tasks performed regardless of the device's configuration, and other tasks that depend on the device configuration.

For more information, see the **show task** command.

Related Documentation

- [Routing Protocol Process Memory FAQs Overview on page 173](#)

PART 5

Index

- [Index on page 183](#)

Index

Symbols

| | |
|--|-----|
| #, comments in configuration statements..... | xii |
| (), in syntax descriptions..... | xii |
| < >, in syntax descriptions..... | xii |
| [], in configuration statements..... | xii |
| { }, in configuration statements..... | xii |
| (pipe), in syntax descriptions..... | xii |

B

| | |
|--|--------|
| BGP See secure interdomain routing | |
| origin AS validation..... | 15, 41 |
| secure interdomain routing..... | 15 |
| BGP in multitopology routing..... | 14 |
| braces, in configuration statements..... | xii |
| brackets | |
| angle, in syntax descriptions..... | xii |
| square, in configuration statements..... | xii |

C

| | |
|--|------|
| comments, in configuration statements..... | xii |
| community statement | |
| Multitopology Routing..... | 68 |
| conventions | |
| text and syntax..... | xi |
| curly braces, in configuration statements..... | xii |
| customer support..... | xiii |
| contacting JTAC..... | xiii |

D

| | |
|------------------|------|
| documentation | |
| comments on..... | xiii |

F

| | |
|--------------------------------------|----------|
| FAQs | |
| routing protocol process memory..... | 173, 174 |
| font conventions..... | xi |
| forwarding table | |
| route entries, displaying..... | 134 |

H

| | |
|--------------------------------|-----|
| hidden routes, displaying..... | 148 |
|--------------------------------|-----|

M

| | |
|--------------------------|------|
| manuals | |
| comments on..... | xiii |
| Multitopology Routing | |
| community statement..... | 68 |
| overview..... | 13 |

O

| | |
|---|--------|
| origin AS validation | |
| BGP..... | 15, 41 |
| OSPF | |
| interfaces, displaying..... | 100 |
| link-state database entries, displaying | |
| version 2..... | 80 |
| version 3..... | 89 |
| neighbors | |
| displaying..... | 106 |
| OSPF in multitopology routing..... | 13 |

P

| | |
|--|-----|
| parentheses, in syntax descriptions..... | xii |
|--|-----|

R

| | |
|----------------------------------|----------|
| rib statement | |
| multitopology routing | |
| static routes..... | 69 |
| routes, displaying | |
| active..... | 113 |
| detailed information..... | 119 |
| hidden..... | 148 |
| in a specific routing table..... | 155 |
| in the forwarding table..... | 134 |
| summary statistics..... | 151 |
| to specified network host..... | 166 |
| routing protocol process memory | |
| FAQ..... | 173, 174 |

S

| | |
|--|----------------|
| secure interdomain routing..... | 15, 41 See BGP |
| show (ospf ospf3) interface command..... | 100 |
| show (ospf ospf3) neighbor command..... | 106 |
| show ospf database command..... | 80 |
| show ospf3 database command..... | 89 |
| show route command..... | 113 |
| show route detail command..... | 119 |
| show route forwarding-table command..... | 134 |
| show route hidden command..... | 148 |
| show route summary command..... | 151 |
| show route table command..... | 155 |

support, technical See technical support
syntax conventions.....xi

T

technical support
 contacting JTAC.....xiii
topologies statement
 multitopology routing.....70
topology statement
 filter-based forwarding
 multitopology routing.....71
 multitopology routing.....72
 OSPF.....73
 OSPF interface.....74
topology-id statement
 multitopology routing.....75
traceroute command.....166

V

validation statement
 usage guidelines.....15, 41