Engineering Simplicity

Junos® OS

RIP User Guide

Published
2023-06-13

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Junos® OS RIP User Guide

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

Table of Contents

About This Guide

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

1

CHAPTER

Overview

RIP and RIPng Overview

IN THIS SECTION

RIP Overview

IN THIS SECTION

RIP is an interior gateway protocol (IGP) that uses a distance-vector algorithm to determine the best route to a destination, using the hop count as the metric.

In a RIP network, each router's forwarding table is distributed among the nodes through the flooding of routing table information. Because topology changes are flooded throughout the network, every node maintains the same list of destinations. Packets are then routed to these destinations based on path-cost calculations done at each node in the network.

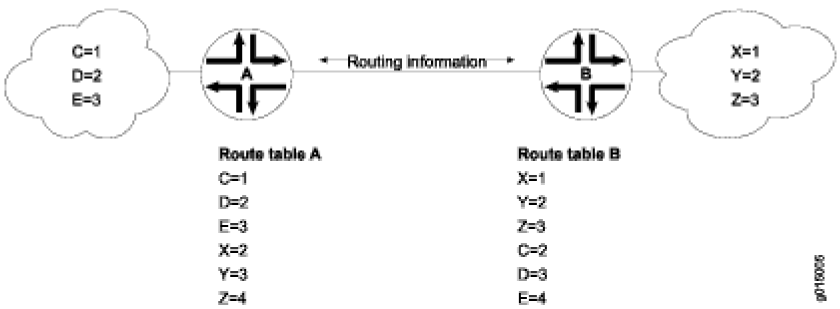
NOTE: In general, the term *RIP* refers to RIP version 1 and RIP version 2.

This topic contains the following sections:

Distance-Vector Routing Protocols

Distance-vector routing protocols transmit routing information that includes a distance vector, typically expressed as the number of hops to the destination. This information is flooded out all protocol-enabled interfaces at regular intervals (every 30 seconds in the case of RIP) to create a network map that is stored in each node's local topology database. Figure 1 on page 3 shows how distance-vector routing works.

Figure 1: Distance-Vector Protocol



In Figure 1 on page 3, Routers A and B have RIP enabled on adjacent interfaces. Router A has known RIP neighbors Routers C, D, and E, which are 1, 2, and 3 hops away, respectively. Router B has known RIP neighbors Routers X, Y, and Z, which are 1, 2, and 3 hops away, respectively. Every 30 seconds, each router floods its entire routing table information out all RIP-enabled interfaces. In this case, flooding exchanges routing table information across the RIP link.

When Router A receives routing information from Router B, it adds 1 to the hop count to determine the new hop count. For example, Router X has a hop count of 1, but when Router A imports the route to X, the new hop count is 2. The imported route also includes information about where the route was learned, so that the original route is imported as a route to Router X through Router B with a hop count of 2.

When multiple routes to the same host are received, RIP uses the distance-vector algorithm to determine which path to import into the forwarding table. The route with the smallest hop count is imported. If there are multiple routes with the same hop count, all are imported into the forwarding table, and traffic is sent along the paths in round-robin fashion.

RIP Protocol Overview

The RIP IGP uses the Bellman-Ford, or *distance-vector*, algorithm to determine the best route to a destination. RIP uses the hop count as the metric. RIP enables hosts and routers to exchange information for computing routes through an IP-based network. RIP is intended to be used as an IGP in reasonably homogeneous networks of moderate size.

The Junos® operating system (Junos OS) supports RIP versions 1 and 2.

> **NOTE**: RIP is not supported for multipoint interfaces.

RIP version 1 packets contain the minimal information necessary to route packets through a network. However, this version of RIP does not support authentication or subnetting.

RIP uses User Datagram Protocol (UDP) port 520.

RIP has the following architectural limitations:

- The longest network path cannot exceed 15 hops (assuming that each network, or hop, has a cost of 1).
- RIP depends on counting to infinity to resolve certain unusual situations—When the network consists of several hundred routers, and when a routing loop has formed, the amount of time and network bandwidth required to resolve a next hop might be great.
- RIP uses only a fixed metric to select a route. Other IGPs use additional parameters, such as measured delay, reliability, and load.

RIP Packets

RIP packets contain the following fields:

- Command—Indicates whether the packet is a request or response message. Request messages seek information for the router’s routing table. Response messages are sent periodically and also when a request message is received. Periodic response messages are called *update messages*. Update messages contain the command and version fields and 25 destinations (by default), each of which includes the destination IP address and the metric to reach that destination.

  > **NOTE**: Beginning with Junos OS Release 11.1, three additional command field types are available to support RIP demand circuits. When you configure an interface for RIP demand circuits, the command field indicates whether the packet is an update request, update response, or update acknowledge message. Neighbor interfaces send updates on demand, not periodically. These command field types are only valid on interfaces configured for RIP demand circuits. For more detailed information, see "RIP Demand Circuits Overview" on page 54.

- Version number—Version of RIP that the originating router is running.
- Address family identifier—Address family used by the originating router. The family is always IP.

- Address—IP address included in the packet.
- Metric—Value of the metric advertised for the address.
- Mask—Mask associated with the IP address (RIP version 2 only).
- Next hop—IP address of the next-hop router (RIP version 2 only).

Routing information is exchanged in a RIP network by RIP request and RIP response packets. A router that has just booted can broadcast a RIP request on all RIP-enabled interfaces. Any routers running RIP on those links receive the request and respond by sending a RIP response packet immediately to the router. The response packet contains the routing table information required to build the local copy of the network topology map.

In the absence of RIP request packets, all RIP routers broadcast a RIP response packet every 30 seconds on all RIP-enabled interfaces. The RIP broadcast is the primary way in which topology information is flooded throughout the network.

Once a router learns about a particular destination through RIP, it starts a timer. Every time it receives a new response packet with information about the destination, the router resets the timer to zero. However, if the router receives no updates about a particular destination for 180 seconds, it removes the destination from its RIP routing table.

In addition to the regular transmission of RIP packets every 30 seconds, if a router detects a new neighbor or detects that an interface is unavailable, it generates a triggered update. The new routing information is immediately broadcast out all RIP-enabled interfaces, and the change is reflected in all subsequent RIP response packets.

Maximizing Hop Count

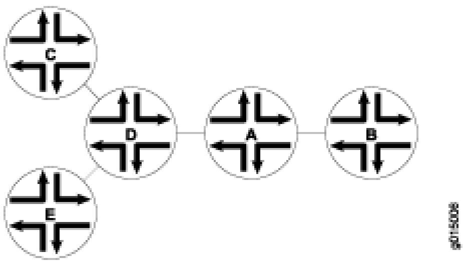
The successful routing of traffic across a RIP network requires that every node in the network maintain the same view of the topology. Topology information is broadcast between RIP neighbors every 30 seconds. If Router A is many hops away from a new host, Router B, the route to B might take significant time to propagate through the network and be imported into Router A's routing table. If the two routers are 5 hops away from each other, Router A cannot import the route to Router B until 2.5 minutes after Router B is online (30 seconds per hop). For large numbers of hops, the delay becomes prohibitive. To help prevent this delay from growing arbitrarily large, RIP enforces a maximum hop count of 15 hops. Any prefix that is more than 15 hops away is treated as unreachable and assigned a hop count equal to infinity. This maximum hop count is called the *network diameter*.

Split Horizon and Poison Reverse Efficiency Techniques

Because RIP functions by periodically flooding the entire routing table out to the network, it generates a lot of traffic. The split horizon and poison reverse techniques can help reduce the amount of network traffic originated by RIP hosts and make the transmission of routing information more efficient.

If a router receives a set of route advertisements on a particular interface, RIP determines that those advertisements do not need to be retransmitted out the same interface. This technique, known as *split horizon*, helps limit the amount of RIP routing traffic by eliminating information that other neighbors on that interface have already learned. Figure 2 on page 6 shows an example of the split horizon technique.

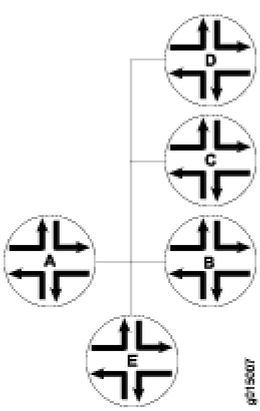
Figure 2: Split Horizon Example



In Figure 2 on page 6, Router A advertises routes to Routers C, D, and E to Router B. In this example, Router A can reach Router C in 2 hops. When Router A advertises the route to Router B, Router B imports it as a route to Router C through Router A in 3 hops. If Router B then readvertised this route to Router A, Router A would import it as a route to Router C through Router B in 4 hops. However, the advertisement from Router B to Router A is unnecessary, because Router A can already reach the route in 2 hops. The split horizon technique helps reduce extra traffic by eliminating this type of route advertisement.

Similarly, the poison reverse technique helps to optimize the transmission of routing information and improve the time to reach network convergence. If Router A learns about unreachable routes through one of its interfaces, it advertises those routes as unreachable (hop count of 16) out the same interface. Figure 3 on page 6 shows an example of the poison reverse technique.

Figure 3: Poison Reverse Example

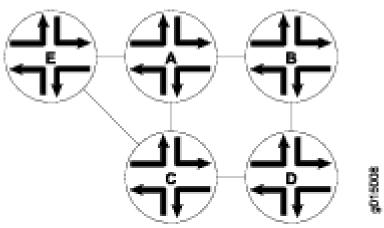


In Figure 3 on page 6, Router A learns through one of its interfaces that routes to Routers C, D, and E are unreachable. Router A readvertises those routes out the same interface as unreachable. The advertisement informs Router B that Routers C, D, and E are definitely not reachable through Router A.

Limitations of Unidirectional Connectivity

Because RIP processes routing information based solely on the receipt of routing table updates, it cannot ensure bidirectional connectivity. As Figure 4 on page 7 shows, RIP networks are limited by their unidirectional connectivity.

Figure 4: Limitations of Unidirectional Connectivity



In Figure 4 on page 7, Routers A and D flood their routing table information to Router B. Because the path to Router E has the fewest hops when routed through Router A, that route is imported into Router B's forwarding table. However, suppose that Router A can transmit traffic but is not receiving traffic from Router B because of an unavailable link or invalid routing policy. If the only route to Router E is through Router A, any traffic destined for Router A is lost, because bidirectional connectivity was never established.

OSPF establishes bidirectional connectivity with a three-way handshake.

SEE ALSO

RIP Configuration Overview | 12
Example: Configuring RIP

RIPng Overview

IN THIS SECTION

The Routing Information Protocol next generation (RIPng) is an interior gateway protocol (IGP) that uses a distance-vector algorithm to determine the best route to a destination, using hop count as the metric. RIPng exchanges routing information used to compute routes and is intended for IP version 6 (IPv6)-based networks. RIPng is disabled by default.

On devices in secure context, IPv6 is disabled. You must enable IPv6 to use RIPng. For instructions, see the *Junos OS Interfaces Configuration Guide for Security Devices*.

This topic contains the following sections:

RIPng Protocol Overview

The RIPng IGP uses the Bellman-Ford distance-vector algorithm to determine the best route to a destination, using hop count as the metric. RIPng allows hosts and routers to exchange information for computing routes through an IP-based network. RIPng is intended to act as an IGP for moderately-sized autonomous systems.

RIPng is a distinct routing protocol from RIPv2. The Junos OS implementation of RIPng is similar to RIPv2, but has the following differences:

- RIPng does not need to implement authentication on packets.
- Junos OS does not support multiple instances of RIPng.
- Junos OS does not support RIPng routing table groups.

RIPng is a UDP-based protocol and uses UDP port 521.

RIPng has the following architectural limitations:

- The longest network path cannot exceed 15 hops (assuming that each network, or hop, has a cost of 1).

- RIPng is prone to routing loops when the routing tables are reconstructed. Especially when RIPng is implemented in large networks that consist of several hundred routers, RIPng might take an extremely long time to resolve routing loops.

- RIPng uses only a fixed metric to select a route. Other IGPs use additional parameters, such as measured delay, reliability, and load.

RIPng Standards

RIPng is defined in the following documents:

- RFC 2080, *RIPng for IPv6*

- RFC 2081, *RIPng Protocol Applicability Statement*

To access Internet Requests for Comments (RFCs) and drafts, see the Internet Engineering Task Force (IETF) website.

RIPng Packets

A RIPng packet header contains the following fields:

- Command—Indicates whether the packet is a request or response message. Request messages seek information for the router's routing table. Response messages are sent periodically or when a request message is received. Periodic response messages are called update messages. Update messages contain the command and version fields and a set of destinations and metrics.

- Version number—Specifies the version of RIPng that the originating router is running. This is currently set to Version 1.

The rest of the RIPng packet contains a list of routing table entries consisting of the following fields:

- Destination prefix—128-bit IPv6 address prefix for the destination.

- Prefix length—Number of significant bits in the prefix.

- Metric—Value of the metric advertised for the address.

- Route tag—A route attribute that must be advertised and redistributed with the route. Primarily, the route tag distinguishes external RIPng routes from internal RIPng routes when routes must be redistributed across an exterior gateway protocol (EGP).

SEE ALSO

Example: Configuring a Basic RIPng Network | 134

Supported RIP and RIPng Standards

Junos OS substantially supports the following RFCs, which define standards for RIP (for IP version 4 [IPv4]) and RIP next generation (RIPng, for IP version 6 [IPv6]).

Junos OS supports authentication for all RIP protocol exchanges (MD5 or simple authentication).

- RFC 1058, *Routing Information Protocol*
- RFC 2080, *RIPng for IPv6*
- RFC 2082, *RIP-2 MD5 Authentication*

  Multiple keys using distinct key IDs are not supported.
- RFC 2453, *RIP Version 2*

The following RFC does not define a standard, but provides information about RIPng. The IETF classifies it as "Informational."

- RFC 2081, *RIPng Protocol Applicability Statement*

SEE ALSO

Supported IPv4, TCP, and UDP Standards
Supported IPv6 Standards
Accessing Standards Documents on the Internet

2

CHAPTER

Configuring RIP

Basic RIP Configuration

IN THIS SECTION

RIP Configuration Overview

To achieve basic connectivity between all RIP hosts in a RIP network, you enable RIP on every interface that is expected to transmit and receive RIP traffic, as described in the steps that follow.

To configure a RIP network:

1. Configure network interfaces. See the *Junos OS Interfaces Configuration Guide for Security Devices*.
2. Define RIP groups, which are logical groupings of interfaces, and add interfaces to the groups. Then, configure a routing policy to export directly connected routes and routes learned through RIP routing exchanges. See "Example: Configuring a Basic RIP Network" on page 13.
3. (Optional) Configure metrics to control traffic through the RIP network. See "Example: Controlling Traffic in a RIP Network with an Incoming Metric" on page 85 and "Example: Controlling Traffic in a RIP Network with an Outgoing Metric" on page 87.
4. (Optional) Configure authentication to ensure that only trusted routers participate in the autonomous system's routing. See "Enabling Authentication with Plain-Text Passwords" on page 24 and Enabling Authentication with MD5 Authentication (CLI Procedure).

SEE ALSO

RIP Overview | 2

Verifying a RIP Configuration | 216

Understanding Basic RIP Routing

RIP is an interior gateway protocol (IGP) that routes packets within a single autonomous system (AS). By default, RIP does not advertise the subnets that are directly connected through the device's interfaces. For traffic to pass through a RIP network, you must create a routing policy to export these routes. Advertising only the direct routes propagates the routes to the immediately adjacent RIP-enabled router only. To propagate all routes through the entire RIP network, you must configure the routing policy to export the routes learned through RIP.

SEE ALSO

RIP Overview | 2

Example: Configuring a Basic RIP Network

IN THIS SECTION

This example shows how to configure a basic RIP network.

Requirements

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

Overview

IN THIS SECTION

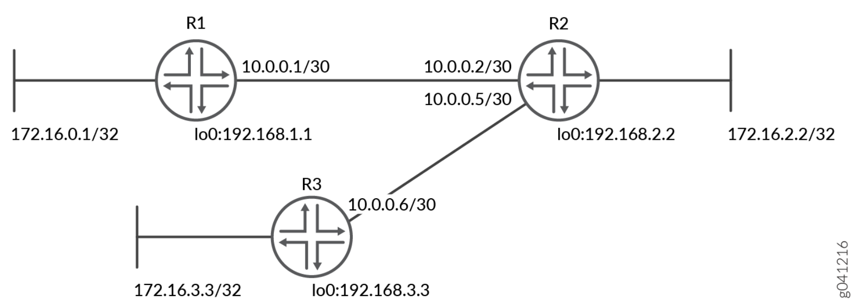
In this example, you configure a basic RIP network, create a RIP group called **rip-group**, and add the directly connected interfaces to the RIP group. Then you configure a routing policy to advertise direct routes using policy statement **advertise-routes-through-rip**.

By default, Junos OS does not advertise RIP routes, not even routes that are learned through RIP. To advertise RIP routes, you must configure and apply an export routing policy that advertises RIP-learned and direct routes.

In Junos OS, you do not need to configure the RIP version. RIP version 2 is used by default.

To use RIP on the device, you must configure RIP on all of the RIP interfaces within the network. Figure 5 on page 14 shows the topology used in this example.

Figure 5: Sample RIP Network Topology



"CLI Quick Configuration" on page 15 shows the configuration for all of the devices in Figure 5 on page 14. The section "No Link Title" on page 16 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces lo0 unit 1 family inet address 172.16.0.1/32
set interfaces lo0 unit 1 family inet address 192.168.1.1/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.1
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 5 family inet address 10.0.0.5/30
set interfaces lo0 unit 2 family inet address 192.168.2.2/32
set interfaces lo0 unit 2 family inet address 172.16.2.2/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.2
set protocols rip group rip-group neighbor fe-1/2/1.5
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set interfaces lo0 unit 3 family inet address 192.168.3.3/32
set interfaces lo0 unit 3 family inet address 172.16.3.3/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.6
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
```

```
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

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 a basic RIP network:

1. Configure the network interfaces.

   This example shows multiple loopback interface addresses to simulate attached networks.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 family inet address 10.0.0.1/30
   user@R1# set lo0 unit 1 family inet address 172.16.0.1/32
   user@R1# set lo0 unit 1 family inet address 192.168.1.1/32
   ```

2. Create the RIP group and add the interface.

   To configure RIP in Junos OS, you must configure a group that contains the interfaces on which RIP is enabled. You do not need to enable RIP on the loopback interface.

   ```
   [edit protocols rip group rip-group]
   user@R1# set neighbor fe-1/2/0.1
   ```

3. Create the routing policy to advertise both direct and RIP-learned routes.

   ```
   [edit policy-options policy-statement advertise-routes-through-rip term 1]
   user@R1# set from protocol direct
   user@R1# set from protocol rip
   user@R1# set then accept
   ```

4. Apply the routing policy.

In Junos OS, you can only apply RIP export policies at the group level.

```
[edit protocols rip group rip-group]
user@R1# set export advertise-routes-through-rip
```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        family inet {
            address 10.0.0.1/30;
        }
    }
}
lo0 {
    unit 1 {
        family inet {
            address 172.16.0.1/32;
            address 192.168.1.1/32;
        }
    }
}
```

```
user@R1# show protocols
rip {
    group rip-group {
        export advertise-routes-through-rip;
        neighbor fe-1/2/0.1;
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-rip {
```

```
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Checking the Routing Table

Purpose

Verify that the routing table is populated with the expected routes..

Action

From operational mode, enter the `show route protocol rip` command.

```
user@R1> show route protocol rip
inet.0: 10 destinations, 10 routes (10 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.4/30        *[RIP/100] 00:59:15, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
```

```
172.16.2.2/32      *[RIP/100] 02:52:48, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
172.16.3.3/32      *[RIP/100] 00:45:05, metric 3, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
192.168.2.2/32     *[RIP/100] 02:52:48, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
192.168.3.3/32     *[RIP/100] 00:45:05, metric 3, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
224.0.0.9/32       *[RIP/100] 00:45:09, metric 1
                         MultiRecv
```

Meaning

The output shows that the routes have been learned from Device R2 and Device R3.

If you were to delete the **from protocol rip** condition in the routing policy on Device R2, the remote routes from Device R3 would not be learned on Device R1.

Looking at the Routes That Device R1 Is Advertising to Device R2

Purpose

Verify that Device R1 is sending the expected routes.

Action

From operational mode, enter the `show route advertising-protocol rip` command.

```
user@R1> show route advertising-protocol rip 10.0.0.1
inet.0: 10 destinations, 10 routes (10 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.0.1/32      *[Direct/0] 05:18:26
                    >    via lo0.1
192.168.1.1/32     *[Direct/0] 05:18:25
                    >    via lo0.1
```

Meaning

Device R1 is sending routes to its directly connected networks.

Looking at the Routes That Device R1 Is Receiving from Device R2

Purpose

Verify that Device R1 is receiving the expected routes.

Action

From operational mode, enter the `show route receive-protocol rip` command.

```
user@R1> show route receive-protocol rip 10.0.0.2
inet.0: 10 destinations, 10 routes (10 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.4/30        *[RIP/100] 02:31:22, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
172.16.2.2/32      *[RIP/100] 04:24:55, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
172.16.3.3/32      *[RIP/100] 02:17:12, metric 3, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
192.168.2.2/32     *[RIP/100] 04:24:55, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
192.168.3.3/32     *[RIP/100] 02:17:12, metric 3, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
```

Meaning

Device R1 is receiving from Device R2 all of Device R2's directly connected networks. Device R1 is also receiving from Device R2 all of Device R3's directly connected networks, which Device R2 learned from Device R3 through RIP.

Verifying the RIP-Enabled Interfaces

Purpose

Verify that all RIP-enabled Interfaces are available and active.

Action

From operational mode, enter the `show rip neighbor` command.

```
user@R1> show rip neighbor
                  Local  Source          Destination     Send   Receive   In
Neighbor          State  Address         Address         Mode   Mode     Met
--------          -----  -------         -----------     ----   -------  ---
fe-1/2/0.1           Up 10.0.0.1        224.0.0.9       mcast  both     1
```

Meaning

The output shows that the RIP-enabled interface on Device R1 is operational.

In general for this command, the output shows a list of the RIP neighbors that are configured on the device. Verify the following information:

- Each configured interface is present. Interfaces are listed in alphabetical order.
- Each configured interface is up. The state of the interface is listed in the **Local State** column. A state of **Up** indicates that the link is passing RIP traffic. A state of **Dn** indicates that the link is not passing RIP traffic. In a point-to-point link, this state generally means that either the end point is not configured for RIP or the link is unavailable.

Verifying the Exchange of RIP Messages

Purpose

Verify that RIP messages are being sent and received on all RIP-enabled interfaces.

Action

From operational mode, enter the `show rip statistics` command.

```
user@R1> show rip statistics
RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              5              0              0              0

fe-1/2/0.1:  5 routes learned; 2 routes advertised; timeout 180s; update interval 30s
Counter                         Total   Last 5 min  Last minute
```

```
-------                    -----------  -----------  -----------
Updates Sent                      2669           10            2
Triggered Updates Sent               2            0            0
Responses Sent                       0            0            0
Bad Messages                         0            0            0
RIPv1 Updates Received               0            0            0
RIPv1 Bad Route Entries              0            0            0
RIPv1 Updates Ignored                0            0            0
RIPv2 Updates Received            2675           11            2
RIPv2 Bad Route Entries              0            0            0
RIPv2 Updates Ignored                0            0            0
Authentication Failures              0            0            0
RIP Requests Received                0            0            0
RIP Requests Ignored                 0            0            0
none                                 0            0            0
```

Meaning

The output shows the number of RIP routes learned. It also shows the number of RIP updates sent and received on the RIP-enabled interfaces. Verify the following information:

- The number of RIP routes learned matches the number of expected routes learned. Subnets learned by direct connectivity through an outgoing interface are not listed as RIP routes.
- RIP updates are being sent on each RIP-enabled interface. If no updates are being sent, the routing policy might not be configured to export routes.
- RIP updates are being received on each RIP-enabled interface. If no updates are being received, the routing policy might not be configured to export routes on the host connected to that subnet. The lack of updates might also indicate an authentication error.

Verifying Reachability of All Hosts in the RIP Network

Purpose

Use the `traceroute` command on each loopback address in the network to verify that all hosts in the RIP network are reachable from each Juniper Networks device.

Action

From operational mode, enter the `traceroute` command.

```
user@R1> traceroute 192.168.3.3
traceroute to 192.168.3.3 (192.168.3.3), 30 hops max, 40 byte packets
 1  10.0.0.2 (10.0.0.2)  1.094 ms  1.028 ms  0.957 ms
 2  192.168.3.3 (192.168.3.3)  1.344 ms  2.245 ms  2.125 ms
```

Meaning

Each numbered row in the output indicates a routing hop in the path to the host. The three-time increments indicate the round-trip time (RTT) between the device and the hop for each traceroute packet.

To ensure that the RIP network is healthy, verify the following information:

- The final hop in the list is the host you want to reach.
- The number of expected hops to the host matches the number of hops in the traceroute output. The appearance of more hops than expected in the output indicates that a network segment is probably unreachable. It might also indicate that the incoming or outgoing metric on one or more hosts has been set unexpectedly.

RIP Authentication

IN THIS SECTION

Understanding RIP Authentication

RIPv2 provides authentication support so that RIP links can require authentication keys (passwords) before they become active. Authentication provides an additional layer of security on the network beyond the other security features. By default, this authentication is disabled.

Authentication keys can be specified in either plain-text or MD5 form. Authentication requires all routers within the RIP network or subnetwork to have the same authentication type and key (password) configured.

This type of authentication is not supported on RIPv1 networks.

MD5 authentication uses an encoded MD5 checksum that is included in the transmitted packet. For MD5 authentication to work, both the receiving and transmitting routing devices must have the same MD5 key. You define an MD5 key for each interface. If MD5 is enabled on an interface, that interface accepts routing updates only if MD5 authentication succeeds. Otherwise, updates are rejected. The routing device only accepts RIPv2 packets sent using the same key identifier (ID) that is defined for that interface. Starting in Junos OS Release 20.3R1, we support multiple MD5 authentication keys for RIPv2 for increased security. This supports adding of MD5 keys with their `start-time`. RIPv2 packets are transmitted with MD5 authentication using the first configured key. RIPv2 authentication switches to the next key based on its configured key `start-time`. This provides automatic key switching without user intervention to change the MD5 keys as in the case of having only one MD5 key.

Note that the RIPv2 authentication described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

SEE ALSO

RIP Overview | 2

Enabling Authentication with Plain-Text Passwords

To configure authentication that requires a plain-text password to be included in the transmitted packet, enable simple authentication by performing these steps on all RIP devices in the network:

1. Navigate to the top of the configuration hierarchy.
2. Perform the configuration tasks described in Table 1 on page 25.
3. If you are finished configuring the router, commit the configuration.

Table 1: Configuring Simple RIP Authentication

Task	CLI Configuration Editor
Navigate to **Rip** level in the configuration hierarchy.	From the `[edit]` hierarchy level, enter **edit protocols rip**
Set the authentication type to **simple**.	Set the authentication type to **simple**: `set authentication-type simple`
Set the authentication key to a simple-text password. The password can be from 1 through 16 contiguous characters long and can include any ASCII strings.	Set the authentication key to a simple-text password: `set authentication-key` *`password`*

SEE ALSO

RIP Configuration Overview | 12

Example: Configuring Route Authentication for RIP using single MD5 key

IN THIS SECTION

This example shows how to configure authentication for a RIP network.

Requirements

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

Overview

IN THIS SECTION

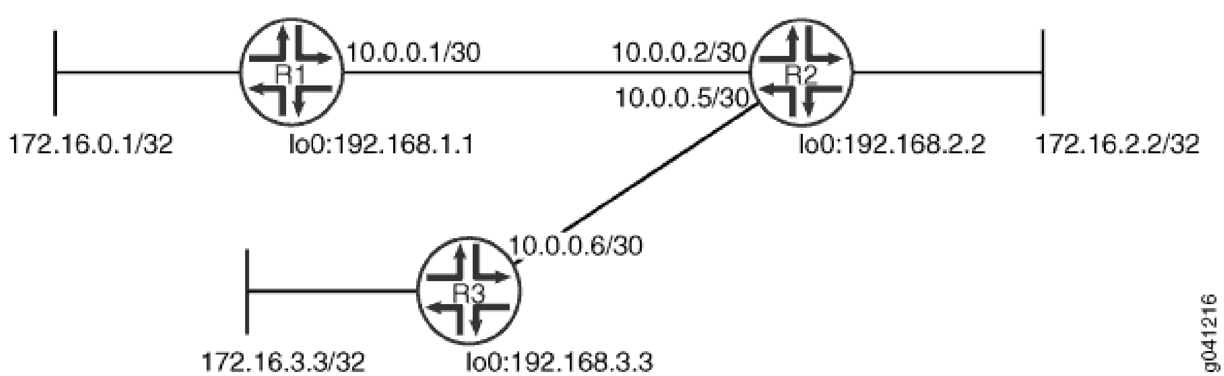
You can configure the router to authenticate RIP route queries. By default, authentication is disabled. You can use one of the following authentication methods:

- Simple authentication—Uses a text password that is included in the transmitted packet. The receiving router uses an authentication key (password) to verify the packet.
- MD5 authentication—Creates an encoded checksum that is included in the transmitted packet. The receiving router uses an authentication key (password) to verify the packet's MD5 checksum.

This example shows MD5 authentication.

Figure 6 on page 26 shows the topology used in this example.

Figure 6: RIP Authentication Network Topology



"CLI Quick Configuration" on page 27 shows the configuration for all of the devices in Figure 6 on page 26. The section "No Link Title" on page 28 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces lo0 unit 1 family inet address 172.16.0.1/32
set interfaces lo0 unit 1 family inet address 192.168.1.1/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.1
set protocols rip authentication-type md5
set protocols rip authentication-key "$ABC123$ABC123"
set protocols rip traceoptions file rip-authentication-messages
set protocols rip traceoptions flag auth
set protocols rip traceoptions flag packets
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 5 family inet address 10.0.0.5/30
set interfaces lo0 unit 2 family inet address 192.168.2.2/32
set interfaces lo0 unit 2 family inet address 172.16.2.2/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.2
```

```
set protocols rip group rip-group neighbor fe-1/2/1.5
set protocols rip authentication-type md5
set protocols rip authentication-key "$ABC123$ABC123"
set protocols rip traceoptions file rip-authentication-messages
set protocols rip traceoptions flag auth
set protocols rip traceoptions flag packets
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set interfaces lo0 unit 3 family inet address 192.168.3.3/32
set interfaces lo0 unit 3 family inet address 172.16.3.3/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.6
set protocols rip authentication-type md5
set protocols rip authentication-key "$ABC123$ABC123"
set protocols rip traceoptions file rip-authentication-messages
set protocols rip traceoptions flag auth
set protocols rip traceoptions flag packets
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

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 RIP authentication:

1. Configure the network interfaces.

   This example shows multiple loopback interface addresses to simulate attached networks.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 family inet address 10.0.0.1/30
   ```

```
user@R1# set lo0 unit 1 family inet address 172.16.0.1/32
user@R1# set lo0 unit 1 family inet address 192.168.1.1/32
```

2. Create the RIP group and add the interface.

   To configure RIP in Junos OS, you must configure a group that contains the interfaces on which RIP is enabled. You do not need to enable RIP on the loopback interface.

```
[edit protocols rip group rip-group]
user@R1# set neighbor fe-1/2/0.1
```

3. Create the routing policy to advertise both direct and RIP-learned routes.

```
[edit policy-options policy-statement advertise-routes-through-rip term 1]
user@R1# set from protocol direct
user@R1# set from protocol rip
user@R1# set then accept
```

4. Apply the routing policy.

   In Junos OS, you can only apply RIP export policies at the group level.

```
[edit protocols rip group rip-group]
user@R1# set export advertise-routes-through-rip
```

5. Require MD5 authentication for RIP route queries received on an interface.

   The passwords must match on neighboring RIP routers. If the password does not match, the packet is rejected. The password can be from 1 through 16 contiguous characters long and can include any ASCII strings.

   Do not enter the password as shown here. The password shown here is the encrypted password that is displayed in the configuration after the actual password is already configured.

```
[edit protocols rip]
user@R1# set authentication-type md5
user@R1# set authentication-key "$ABC123$ABC123"
```

6. Configure tracing operations to track authentication.

```
[edit protocols rip traceoptions]
user@R1# set file rip-authentication-messages
user@R1# set flag auth
user@R1# set flag packets
```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        family inet {
            address 10.0.0.1/30;
        }
    }
}
lo0 {
    unit 1 {
        family inet {
            address 172.16.0.1/32;
            address 192.168.1.1/32;
        }
    }
}
```

```
user@R1# show protocols
rip {
    traceoptions {
        file rip-authentication-messages;
        flag auth;
        flag packets;
    }
    authentication-type md5;
    authentication-key $ABC123$ABC123; ## SECRET-DATA
```

```
    group rip-group {
        export advertise-routes-through-rip;
        neighbor fe-1/2/0.1;
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-rip {
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Checking for Authentication Failures

Purpose

Verify that there are no authentication failures.

Action

From operational mode, enter the `show rip statistics` command.

```
user@R1> show rip statistics
RIPv2 info: port 520; holddown 120s.
```

```
    rts learned  rts held down  rqsts dropped  resps dropped
              5              0              0              0

fe-1/2/0.1:  5 routes learned; 2 routes advertised; timeout 180s; update interval 30s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                     2669           10            2
Triggered Updates Sent              2            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received           2675           11            2
RIPv2 Bad Route Entries             0            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               0            0            0
RIP Requests Ignored                0            0            0
none                                0            0            0
```

Meaning

The output shows that there are no authentication failures.

Verifying That MD5 Authentication Is Enabled in RIP Update Packets

Purpose

Use tracing operations to verify that MD5 authentication is enabled in RIP updates.

Action

From operational mode, enter the `show log` command.

```
user@R1> show log rip-authentication-messages | match md5
Feb 15 15:45:13.969462          sending msg 0xb9a8c04, 3 rtes (needs MD5)
Feb 15 15:45:43.229867          sending msg 0xb9a8c04, 3 rtes (needs MD5)
Feb 15 15:46:13.174410          sending msg 0xb9a8c04, 3 rtes (needs MD5)
Feb 15 15:46:42.716566          sending msg 0xb9a8c04, 3 rtes (needs MD5)
```

```
Feb 15 15:47:11.425076          sending msg 0xb9a8c04, 3 rtes (needs MD5)
...
```

Meaning

The **(needs MD5)** output shows that all route updates require MD5 authentication.

SEE ALSO

Understanding Basic RIP Routing | 13

Example: Configuring Route Authentication for RIP using multiple MD5 keys

IN THIS SECTION

This example shows how to configure authentication for a RIP network using multiple MD5 keys and how to configure a transition of MD5 keys on a RIP interface.

Requirements

This example uses the following hardware and software components:.

- Three ACX Series routers
- Junos OS Release 20.3 or later

Overview

MD5 authentication uses an encoded MD5 checksum that is included in the transmitted packet. For MD5 authentication to work, both the receiving and transmitting routing devices must have the same MD5 key.

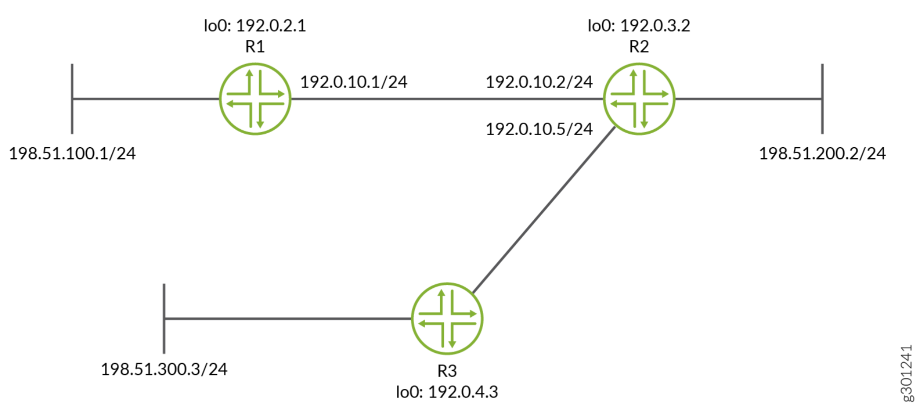
You define an MD5 key for each interface. If MD5 is enabled on an interface, that interface accepts routing updates only if MD5 authentication succeeds. Otherwise, updates are rejected. The routing device only accepts RIPv2 packets sent using the same key identifier (ID) that is defined for that interface.

For increased security, you can configure multiple MD5 keys, each with a unique key ID, and set the date and time to switch to a new key. The receiver of the RIPv2 packet uses the ID to determine which key to use for authentication. RIPv2 with multiple MD5 key feature supports adding of MD5 keys with their start-time. RIPv2 packets are transmitted with MD5 authentication using the first configured key. RIPv2 authentication switches to the next key based on its configured respective key start-time. This provides automatic key switching without user intervention to change the MD5 keys as in case of having only one MD5 key.

This example shows RIPv2 mutliple MD5 keys authentication.

Figure 7 on page 34 shows the topology used in this example.

Figure 7: Network Topology for RIP Authentication using multiple MD5 keys



"CLI Quick Configuration" on page 35 shows the configuration for all of the devices in Figure 7 on page 34. The section "CLI Quick Configuration" on page 35 describes the steps on Device R1.

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces lo0 unit 1 family inet address 172.16.0.1/32
set interfaces lo0 unit 1 family inet address 192.168.1.1/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.1
set protocols rip authentication-selective-md5 2 key $ABC123$ABC123 start-time 2020-02-01.01:01
set protocols rip authentication-selective-md5 3 key $MNO123$MNO123 start-time 2020-03-02.02:01
set protocols rip authentication-selective-md5 4 key $XYZ123$XYZ123 start-time 2020-04-03.03:01
set protocols rip traceoptions file rip-authentication-messages
set protocols rip traceoptions flag auth
set protocols rip traceoptions flag packets
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 5 family inet address 10.0.0.5/30
set interfaces lo0 unit 2 family inet address 192.168.2.2/32
set interfaces lo0 unit 2 family inet address 172.16.2.2/32
set protocols rip group rip-group export advertise-routes-through-rip
```

```
set protocols rip group rip-group neighbor fe-1/2/0.2
set protocols rip group rip-group neighbor fe-1/2/1.5
set protocols rip authentication-type md5
set protocols rip authentication-selective-md5 2 key $ABC123$ABC123 start-time 2020-02-01.01:01
set protocols rip authentication-selective-md5 3 key $MNO123$MNO123 start-time 2020-03-02.02:01
set protocols rip authentication-selective-md5 4 key $XYZ123$XYZ123 start-time 2020-04-03.03:01
set protocols rip traceoptions file rip-authentication-messages
set protocols rip traceoptions flag auth
set protocols rip traceoptions flag packets
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set interfaces lo0 unit 3 family inet address 192.168.3.3/32
set interfaces lo0 unit 3 family inet address 172.16.3.3/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.6
set protocols rip authentication-selective-md5 2 key $ABC123$ABC123 start-time 2020-02-01.01:01
set protocols rip authentication-selective-md5 3 key $MNO123$MNO123 start-time 2020-03-02.02:01
set protocols rip authentication-selective-md5 4 key $XYZ123$XYZ123 start-time 2020-04-03.03:01
set protocols rip traceoptions file rip-authentication-messages
set protocols rip traceoptions flag auth
set protocols rip traceoptions flag packets
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

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 RIP authentication:

1. Configure the network interfaces.

This example shows multiple loopback interface addresses to simulate attached networks.

```
[edit interfaces]
user@R1# set fe-1/2/0 unit 1 family inet address 192.0.10.1/24
user@R1# set lo0 unit 1 family inet address 198.51.100.1/24
user@R1# set lo0 unit 1 family inet address 192.0.2.1/32
```

2. Create the RIP group and add the interface.

   To configure RIP in Junos OS, you must configure a group that contains the interfaces on which RIP is enabled. You do not need to enable RIP on the loopback interface.

```
[edit protocols rip group rip-group]
user@R1# set neighbor fe-1/2/0
```

3. Create the routing policy to advertise both direct and RIP-learned routes.

```
[edit policy-options policy-statement advertise-routes-through-rip term 1]
user@R1# set from protocol direct
user@R1# set from protocol rip
user@R1# set then accept
```

4. Apply the routing policy.

   In Junos OS, you can only apply RIP export policies at the group level.

```
[edit protocols rip group rip-group]
user@R1# set export advertise-routes-through-rip
```

5. You can configure multiple MD5 keys by using different Key IDs. The key-IDs must match with the key-IDs of the neighboring RIP routers. If a router receives a packet with a key-id that is not within its configured set of keys, then the packet is rejected and is considered as authentication failure.

   The key-ID can be a number from 0 to 255 which uniquely identifies an MD5 key and the key value can be an ASCII string upto 16 characters long.

Do not enter the password as shown here. The password shown here is the encrypted password that is displayed in the configuration after the actual password is already configured.

```
[edit protocols rip]
user@R1# set authentication-selective-md5 key-id key key-value start-time time
user@R1# set authentication-selective-md5 2 key $ABC123$ABC123 start-time 2020-02-01.01:01
```

The `authentication-selective-md5` can be repeated to configure multiple keys.

6. If you want to migrate from an existing md5 authentication key, then you can configure another key with a start-time in future with enough leeway so as to allow configuring all the routers on the link. The transition to the new key is based on its start-time and it happens as soon as the clock reaches the start-time. You may delete keys that are no longer valid by entering the following command:.

```
[edit protocols rip]
user@host# delete authentication-selective-md5 key-id
```

NOTE: The start time is relevant for transmission only and not for receiving RIPv2 packets. Acceptance of received packets is based on the keys configured.

For example, if the time now is February 1, 2020, 1:00 AM and the following key is configured:

```
[edit protocols rip]
user@host# set authentication-selective-md5 2 key $ABC123$ABC123 start-time 2020-02-01.01:01
```

If you want to transition from this key to another key on March 2, at 2:00 AM, and you are able to configure all the routers on the link with the new key at the same time, then you may configure the following key:

```
[edit protocols rip]
user@host# set authentication-selective-md5 3 key $MNO123$MNO123 start-time 2020-03-02.02:01
```

At 2:00 AM, once all the routers switch to the new key, you can safely delete key with id 2 by entering the following command.

```
[edit protocols rip]
user@host# delete authentication-selective-md5 2
```

7. Deletion of active key: If you delete the lastest active key, the system checks for the current configuration and uses the key with the latest key-ID within the existing configuration for RIPv2 packet transmission.

   For example, If you have configured the following keys with the key-ids:

```
[edit protocols rip]
user@R1# set authentication-selective-md5 2 key $ABC123$ABC123 start-time 2020-02-01.01:01
user@R1#set authentication-selective-md5 3 key $MNO123$MNO123 start-time 2020-03-02.02:01
user@R1#set authentication-selective-md5 4 key $XYZ123$XYZ123 start-time 2020-04-03.03:01
```

   The active key in this configuration is the key with key ID 4 and is used for sending the RIPv2 packet out. If you delete the active key ID 4, then the system checks for current configuration and looks for the key with the latest start-time, that is the key with ID 3 and uses it for packet transmission.

8. Configure tracing operations to track authentication.

```
[edit protocols rip traceoptions]
user@R1# set file rip-authentication-messages
user@R1# set flag auth
user@R1# set flag packets
```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        family inet {
            address 192.0.10.1/24;
        }
    }
}
lo0 {
    unit 1 {
        family inet {
            address 198.51.100.1/24;
```

```
            address 192.0.2.1/32;
        }
    }
}
```

```
user@R1# show protocols
rip {
    traceoptions {
        file rip-authentication-messages;
        flag auth;
        flag packets;
    }
    authentication-selective-md5 2 key $ABC123$ABC123 start-time 2020-02-01.01:01 ## SECRET-DATA
    authentication-selective-md5 3 key $MNO123$MNO123 start-time 2020-03-02.02:01 ## SECRET-DATA
    authentication-selective-md5 4 key $XYZ123$XYZ123 start-time 2020-04-03.03:01 ## SECRET-DATA
    group rip-group {
        export advertise-routes-through-rip;
        neighbor ge-0/0/5.0;
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-rip {
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Checking for Authentication Failures

Purpose

To check for authentication failures counters.

Action

From operational mode, enter the `show rip statistics` command.

```
user@R1> show rip statistics
RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              5              0              0              0

ge-0/0/5.0:  5 routes learned; 28 routes advertised; timeout 180s; update interval 30s
Counter                          Total   Last 5 min  Last minute
-------                    -----------  -----------  -----------
Updates Sent                     53058           20            4
Triggered Updates Sent               2            0            0
Responses Sent                       0            0            0
Bad Messages                         0            0            0
RIPv1 Updates Received               0            0            0
RIPv1 Bad Route Entries              0            0            0
RIPv1 Updates Ignored                0            0            0
RIPv2 Updates Received           26538           10            2
RIPv2 Bad Route Entries              0            0            0
RIPv2 Updates Ignored                0            0            0
Authentication Failures                23853            0            0
RIP Requests Received                0            0            0
RIP Requests Ignored                 0            0            0
none                                 0            0            0
```

Meaning

The `Authentication Failures` counter displays the authentication failures count. This output shows that the authentication failure count is 23853.

Checking for the current active MD5 key.

Purpose

To check for the current active key being used.

Action

From operational mode, enter the `show rip neighbor fe-1/2/0` command.

```
user@R1> show rip neighbor fe-1/2/0
                  Local  Source          Destination     Send   Receive   In
Neighbor          State  Address         Address         Mode   Mode      Met
--------          -----  -------         -----------     ----   -------   ---
fe-1/2/0           Up 14.14.14.1      224.0.0.9       mcast  both       1
  Auth type: SELECTIVE-MD5, Active key ID: 2, Start time: 1970 Jan  1 05:30:00 IST
```

Verifying That MD5 Authentication Is Enabled in RIP Update Packets

Purpose

Use tracing operations to verify that MD5 authentication is enabled in RIP updates.

Action

From operational mode, enter the `show log` command.

```
user@R1> show log rip-authentication-messages | match md5
Feb 15 15:45:13.969462          sending msg 0xb9a8c04, 3 rtes (needs MD5)
Feb 15 15:45:43.229867          sending msg 0xb9a8c04, 3 rtes (needs MD5)
Feb 15 15:46:13.174410          sending msg 0xb9a8c04, 3 rtes (needs MD5)
Feb 15 15:46:42.716566          sending msg 0xb9a8c04, 3 rtes (needs MD5)
Feb 15 15:47:11.425076          sending msg 0xb9a8c04, 3 rtes (needs MD5)
...
```

Meaning

The **(needs MD5)** output shows that all route updates require MD5 authentication.

SEE ALSO

Understanding Basic RIP Routing | 13

Release History Table

Release	Description
20.3R1	Starting in Junos OS Release 20.3R1, we support multiple MD5 authentication keys for RIPv2 for increased security
15.1X49	Note that the RIPv2 authentication described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

RIP Timers

IN THIS SECTION

Understanding RIP Timers

RIP uses several timers to regulate its operation.

The update interval is the interval at which routes that are learned by RIP are advertised to neighbors. This timer controls the interval between routing updates. The update interval is set to 30 seconds, by default, with a small random amount of time added when the timer is reset. This added time prevents congestion that can occur if all routing devices update their neighbors simultaneously.

To configure the update time interval, include the `update-interval` statement:

```
update-interval seconds;
```

seconds can be a value from 10 through 60.

You can set a route timeout interval. If a route is not refreshed after being installed in the routing table by the specified time interval, the route is marked as invalid and is removed from the routing table after the hold-down period expires.

To configure the route timeout for RIP, include the `route-timeout` statement:

```
route-timeout seconds;
```

seconds can be a value from 30 through 360. The default value is 180 seconds.

RIP routes expire when either a route timeout limit is met or a route metric reaches infinity, and the route is no longer valid. However, the expired route is retained in the routing table for a specified period so that neighbors can be notified that the route has been dropped. This time period is set by configuring the hold-down timer. Upon expiration of the hold-down timer, the route is removed from the routing table.

To configure the hold-down timer for RIP, include the `holddown` statement:

```
holddown seconds;
```

seconds can be a value from 10 through 180. The default value is 120 seconds.

NOTE: In Junos OS Release 11.1 and later, a retransmission timer is available for RIP demand circuits.

Generally, we recommend against changing the RIP timers, unless the effects of a change are well understood. The route timeout should be at least three times the update interval. Normally, the default values are best left in effect for standard operations.

Example: Configuring RIP Timers

IN THIS SECTION

This example shows how to configure the RIP update interval and how to monitor the impact of the change.

Requirements

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

Overview

IN THIS SECTION

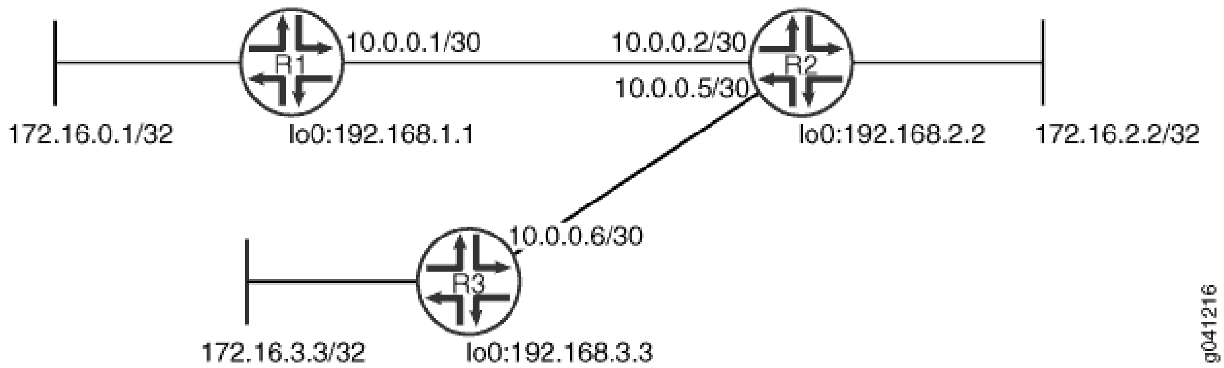
In this example, Device R2 has an update interval of 60 seconds for its neighbor, Device R1, and an update interval of 10 seconds for its neighbor, Device R3.

This example is not necessarily practical, but it is shown for demonstration purposes. Generally, we recommend against changing the RIP timers, unless the effects of a change are well understood. Normally, the default values are best left in effect for standard operations.

An export policy is also shown because an export policy is required as part of the minimum configuration for RIP.

Figure 8 on page 46 shows the topology used in this example.

Figure 8: RIP Timers Network Topology



"CLI Quick Configuration" on page 46 shows the configuration for all of the devices in Figure 8 on page 46. The section "No Link Title" on page 47 describes the steps on Device R2.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces lo0 unit 1 family inet address 172.16.0.1/32
set interfaces lo0 unit 1 family inet address 192.168.1.1/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.1
```

```
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 5 family inet address 10.0.0.5/30
set interfaces lo0 unit 2 family inet address 192.168.2.2/32
set interfaces lo0 unit 2 family inet address 172.16.2.2/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.2 update-interval 60
set protocols rip group rip-group neighbor fe-1/2/1.5 update-interval 10
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set interfaces lo0 unit 3 family inet address 192.168.3.3/32
set interfaces lo0 unit 3 family inet address 172.16.3.3/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.6
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Step-by-Step Procedure

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

To configure the RIP update interval:

1. Configure the network interfaces.

This example shows multiple loopback interface addresses to simulate attached networks.

```
[edit interfaces]
user@R2# set fe-1/2/0 unit 2 family inet address 10.0.0.2/30
user@R2# set fe-1/2/1 unit 5 family inet address 10.0.0.5/30
user@R2# set lo0 unit 2 family inet address 192.168.2.2/32
user@R2# set lo0 unit 2 family inet address 172.16.2.2/32
```

2. Configure different update intervals for the two RIP neighbors.

   To configure RIP in Junos OS, you must configure a group that contains the interfaces on which RIP is enabled. You do not need to enable RIP on the loopback interface.

```
[edit protocols rip group rip-group]
user@R2# set neighbor fe-1/2/0.2 update-interval 60
user@R2# set neighbor fe-1/2/1.5 update-interval 10
```

3. Create the routing policy to advertise both direct and RIP-learned routes.

```
[edit policy-options policy-statement advertise-routes-through-rip term 1]
user@R2# set from protocol direct
user@R2# set from protocol rip
user@R2# set then accept
```

4. Apply the routing policy.

   In Junos OS, you can only apply RIP export policies at the group level.

```
[edit protocols rip group rip-group]
user@R2# set export advertise-routes-through-rip
```

Results

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

```
user@R2# show interfaces
fe-1/2/0 {
    unit 2 {
        family inet {
            address 10.0.0.2/30;
        }
    }
}
fe-1/2/1 {
    unit 5 {
        family inet {
            address 10.0.0.5/30;
        }
    }
}
lo0 {
    unit 2 {
        family inet {
            address 192.168.2.2/32;
            address 172.16.2.2/32;
        }
    }
}
```

```
user@R2# show protocols
rip {
    group rip-group {
        export advertise-routes-through-rip;
        neighbor fe-1/2/0.2 {
            update-interval 60;
        }
        neighbor fe-1/2/1.5 {
            update-interval 10;
        }
```

```
    }
}
```

```
user@R2# show policy-options
policy-statement advertise-routes-through-rip {
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Checking the RIP Updates Sent by Device R2

Purpose

Make sure that the RIP update packets are sent at the expected interval.

Action

From operational mode, enter the `show rip statistics` command.

```
user@R2> show rip statistics
RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              4              2              0              0
```

```
fe-1/2/0.2:  2 routes learned; 5 routes advertised; timeout 180s; update interval 60s
Counter                          Total   Last 5 min  Last minute
-------                    -----------  -----------  -----------
Updates Sent                       123            5            1
Triggered Updates Sent               0            0            0
Responses Sent                       0            0            0
Bad Messages                         0            0            0
RIPv1 Updates Received               0            0            0
RIPv1 Bad Route Entries              0            0            0
RIPv1 Updates Ignored                0            0            0
RIPv2 Updates Received             244           10            2
RIPv2 Bad Route Entries              0            0            0
RIPv2 Updates Ignored                0            0            0
Authentication Failures              0            0            0
RIP Requests Received                0            0            0
RIP Requests Ignored                 0            0            0
none                                 0            0            0

fe-1/2/1.5:  2 routes learned; 5 routes advertised; timeout 180s; update interval 10s
Counter                          Total   Last 5 min  Last minute
-------                    -----------  -----------  -----------
Updates Sent                       734           32            6
Triggered Updates Sent               0            0            0
Responses Sent                       0            0            0
Bad Messages                         0            0            0
RIPv1 Updates Received               0            0            0
RIPv1 Bad Route Entries              0            0            0
RIPv1 Updates Ignored                0            0            0
RIPv2 Updates Received             245           11            2
RIPv2 Bad Route Entries              0            0            0
RIPv2 Updates Ignored                0            0            0
Authentication Failures              0            0            0
RIP Requests Received                0            0            0
RIP Requests Ignored                 0            0            0
none                                 0            0            0
```

Meaning

The **update interval** field shows that the interval is 60 seconds for Neighbor R1 and 10 seconds for Neighbor R3. The **Updates Sent** field shows that Device R2 is sending updates to Device R1 at roughly 1/6 of the rate that it is sending updates to Device R3.

Checking the RIP Updates Received by Device R2

Purpose

Make sure that the RIP update packets are sent at the expected interval.

Action

From operational mode, enter the `show rip statistics` command.

```
user@R1> show rip statistics
RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              5              0              0              0

fe-1/2/0.1:  5 routes learned; 2 routes advertised; timeout 180s; update interval 30s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                      312           10            2
Triggered Updates Sent              2            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received            181            5            1
RIPv2 Bad Route Entries             0            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               1            0            0
RIP Requests Ignored                0            0            0
none                                0            0            0
```

Meaning

The **RIPv2 Updates Received** field shows the number of updates received from Device R2.

Checking the RIP Updates Received by Device R3

Purpose

Make sure that the RIP update packets are sent at the expected interval.

Action

From operational mode, enter the `show rip statistics` command.

```
user@R3> show rip statistics
RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              5              0              0              0

fe-1/2/0.6:  5 routes learned; 2 routes advertised; timeout 180s; update interval 30s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                      314           11            2
Triggered Updates Sent              1            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received            827           31            6
RIPv2 Bad Route Entries             0            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               0            0            0
RIP Requests Ignored                0            0            0
none                                0            0            0
```

Meaning

The **RIPv2 Updates Received** field shows the number of updates received from Device R2.

Release History Table

Release	Description
11.1	In Junos OS Release 11.1 and later, a retransmission timer is available for RIP demand circuits.

RIP Demand Circuits

IN THIS SECTION

RIP Demand Circuits Overview

IN THIS SECTION

RIP periodically sends routing information (RIP packets) to neighboring devices. These periodic broadcasts can consume bandwidth resources and interfere with network traffic by preventing WAN circuits from being closed. Demand circuits for RIP is defined in RFC 2091 and overcomes these issues by exchanging incremental updates on demand.

A demand circuit is a point-to-point connection between two neighboring interfaces configured for RIP. Demand circuits preserve bandwidth by establishing a link when data needs to be transferred, and terminating the link when the data transfer is complete. Demand circuits increase the efficiency of RIP on the configured interfaces by offering minimal network overhead in terms of messages passed between the demand circuit end points, conserving resources, and reducing costs.

By configuring RIP demand circuits, a specific event triggers the device to send an update, thereby eliminating the periodic transmission of RIP packets over the neighboring interface. To save overhead, the device sends RIP information only when changes occur in the routing database, such as:

- The device is first powered on
- The device receives a request for route update information
- A change occurs in the network
- The demand circuit goes down or comes up

The device sends update requests, update responses, and acknowledgments. In addition, the device retransmits updates and requests until valid acknowledgments are received. The device dynamically learns RIP neighbors. If the neighboring interface goes down, RIP flushes routes learned from the neighbor's IP address.

Routes learned from demand circuits do not age like other RIP entries because demand circuits are in a permanent state. Routes in a permanent state are only removed under the following conditions:

- A formerly reachable route changes to unreachable in an incoming response
- The demand circuit is down due to an excessive number of unacknowledged retransmissions

You can also set the *RIP hold-down timer* and the RIP demand circuit retransmission timer to regulate performance. The demand circuit uses these timers to determine if there is a change that requires update messages to be sent. There is also a database timer that runs only when RIP flushes learned routes from the routing table.

This topic includes the following sections:

RIP Demand Circuit Packets

When you configure an interface for RIP demand circuits, the supported command field packet types are different than those for RIP version 1 and RIP version 2. RIP packets for RIP demand circuits contain three additional packet types and an extended 4-byte update header. Both RIP version 1 and RIP version 2 support the three packet types and the extended 4-byte header. Table 2 on page 56 describes the three packet types.

Table 2: RIP Demand Circuit Packet Types

Packet Type	Description
Update Request	Update request messages seek information for the device's routing table. This message is sent when the device is first powered on or when a down demand circuit comes up. The device sends this message every 5 seconds (by default) until an update response message is received.
Update Response	Update response messages are sent in response to an update request message, which occurs when the device is first powered on or when a down demand circuit comes up. Each update response message contains a sequence number that the neighbor uses to acknowledge the update request.
Update Acknowledge	Update acknowledge messages are sent in response to every update response message received by the neighbor.

NOTE: These packets are only valid on interfaces configured for RIP demand circuits. If a demand circuit receives a RIP packet that does not contain these packet types, it silently discards the packet and logs an error message similar to the following:

Ignoring RIP packet with invalid version 0 from neighbor 10.0.0.0 and source 10.0.0.1

Timers Used by RIP Demand Circuits

RIP demand circuits use the RIP hold-down timer and the RIP demand circuit retransmission timer to regulate performance and to determine if there is a change in the network that requires the device to send update messages. The hold-down timer is a global RIP timer that affects the entire RIP configuration; whatever range you configure for RIP applies to RIP demand circuits. The retransmission timer affects only RIP demand circuits. In addition, there is a database timer that runs only when RIP flushes learned routes from the routing table.

- Hold-down timer (global RIP timer)—Use the hold-down timer to configure the number of seconds that RIP waits before updating the routing table. The value of the hold-down timer affects the entire RIP configuration, not just the demand circuit interfaces. The hold-down timer starts when a route timeout limit is met, when a formerly reachable route is unreachable, or when a demand circuit interface is down. When the hold-down timer is running, routes are advertised as unreachable on other interfaces. When the hold-down timer expires, the route is removed from the routing table if all destinations are aware that the route is unreachable or the remaining destinations are down. By default, RIP waits 120 seconds between routing table updates. The range is from 10 to 180 seconds.

- Retransmission timer (RIP demand circuit timer)—RIP demand circuits send update messages every 5 seconds to an unresponsive peer. Use the retransmission timer to limit the number of times a demand circuit resends update messages to an unresponsive peer. If the configured retransmission threshold is reached, routes from the next hop router are marked as unreachable and the hold-down timer starts. The value of the retransmission timer affects only the demand circuit interfaces. To determine the number of times to resend the update message, use the following calculation:

```
5 seconds * number of retransmissions = retransmission seconds
```

  The retransmission range is from 5 through 180 seconds, which corresponds to sending an update message a minimum of 1 time (5 seconds) and a maximum of 36 times (180 seconds).

- Database timer (global timeout timer)—Routes learned from demand circuits do not age like other RIP entries because demand circuits are in a permanent state. On a RIP demand circuit, the database timer starts upon receipt of the update response message with the flush flag sent from a RIP demand circuit peer. When the neighbor receives this message, all routes from that peer are flushed, and the database timer starts and runs for the configured route timeout interval. When the database timer is running, routes are still advertised as reachable on other interfaces. When the database timer expires, the device advertises all routes from its peer as unreachable.

SEE ALSO

RIP Overview | 2
demand-circuit (Protocols RIP) | 234
Example: Configuring RIP Timers | 44
Example: Configuring RIP Demand Circuits | 57
holddown (Protocols RIP) | 244
max-retrans-time | 249

Example: Configuring RIP Demand Circuits

IN THIS SECTION

This example describes how to configure an interface as a RIP demand circuit.

Requirements

Before you begin, configure the device interfaces. See the Junos OS Network Interfaces Library for Routing Devices or the *Junos OS Interfaces Configuration Guide for Security Devices*.

Overview

A demand circuit is a point-to-point connection between two neighboring interfaces configured for RIP. Demand circuits increase the efficiency of RIP on the configured interfaces by eliminating the periodic transmission of RIP packets. Demand circuits preserve bandwidth by establishing a link when data needs to be transferred, and terminating the link when the data transfer is complete. In this example, two devices are connected using SONET/SDH interfaces.

> **NOTE**: When you configure RIP demand circuits, any silent removal of the RIP configuration goes unnoticed by the RIP peer and leads to stale entries in the routing table. To clear the stale entries, deactivate and reactivate RIP on the neighboring devices.

In this example, you configure interface **so-0/1/0** with the following settings:

- **demand-circuit**—Configures the interface as a demand circuit. To complete the demand circuit, you must configure both ends of the pair as demand circuits.

- **max-retrans-time**—RIP demand circuits send update messages every 5 seconds to an unresponsive peer. Use the retransmission timer to limit the number of times a demand circuit resends update messages to an unresponsive peer. If the configured retransmission threshold is reached, routes from the next-hop router are marked as unreachable, and the hold-down timer starts. The value of the retransmission timer affects only the demand circuit interfaces. To determine the number of times to resend the update message, use the following calculation:

  ```
  5 seconds x retransmissions = retransmission seconds
  ```

  For example, if you want the demand circuit to send only two update messages to an unresponsive peer, the calculation is: 5 x 2 = 10. When you configure the retransmission timer, you enter 10 seconds.

The retransmission range is from 5 through 180 seconds, which corresponds to sending an update message a minimum of 1 time (5 seconds) and a maximum of 36 times (180 seconds).

Configuration

IN THIS SECTION

In the following example, you configure a neighboring interface to be a RIP demand circuit and save the configuration.

Procedure

CLI Quick Configuration

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

```
set interfaces so-0/1/0 unit 0 family inet address 192.0.2.0/24
set protocols rip group group1 neighbor so-0/1/0 demand-circuit
set protocols rip group group1 neighbor so-0/1/0 max-retrans-time 10
```

Step-by-Step Procedure

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

To configure a RIP demand circuit on one neighboring interface:

1. Configure the interface.

```
[edit interfaces]
user@host# set so-0/1/0 unit 0 family inet address 192.0.2.0/24
```

2. Configure the neighbor as a demand circuit.

```
[edit protocols rip]
user@host# set group group1 neighbor so-0/1/0 demand-circuit
```

3. Configure the demand circuit retransmission timer.

```
[edit protocols rip]
user@host# set group group1 neighbor so-0/1/0 max-retrans-time 10
```

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

```
[edit]
user@host# commit
```

NOTE: Repeat this entire configuration on the other neighboring interface.

Results

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

```
user@host# show interfaces
so-0/1/0 {
    unit 0 {
        family inet {
            address 192.0.2.0/24;
        }
    }
}
```

```
user@host# show protocols
rip {
    group group1 {
        neighbor so-0/1/0 {
```

```
            demand-circuit;
            max-retrans-time 10;
        }
    }
}
```

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying a Demand Circuit Configuration

Purpose

Verify that the demand circuit configuration is working.

Action

To verify that the demand circuit configuration is in effect, use the **show rip neighbor** operational mode command.

```
user@host> show rip neighbor
                           Source          Destination     Send   Receive   In
Neighbor          State   Address         Address         Mode   Mode      Met
--------          -----   -------         -----------     ----   -------   ---
so-0/1/0.0(DC)       Up   10.10.10.2      224.0.0.9       mcast  both        1
```

When you configure demand circuits, the `show rip neighbor` command displays a DC flag next to the neighboring interface configured for demand circuits.

NOTE: If you configure demand circuits at the `[edit protocols rip group` *`group-name`* `neighbor` *`neighbor-name`*`]` hierarchy level, the output shows only the neighboring interface that you specifically configured as a demand circuit. If you configure demand circuits at the `[edit protocols rip group group-name]` hierarchy level, all of the interfaces in the group are configured as demand circuits. Therefore, the output shows all of the interfaces in that group as demand circuits.

BFD for RIP

IN THIS SECTION

Understanding BFD for RIP

The Bidirectional Forwarding Detection (BFD) Protocol is a simple hello mechanism that detects failures in a network. Hello packets are sent at a specified, regular interval. A neighbor failure is detected when the routing device stops receiving a reply after a specified interval. BFD works with a wide variety of network environments and topologies. BFD failure detection times are shorter than RIP detection times, providing faster reaction times to various kinds of failures in the network. Instead of waiting for the routing protocol neighbor timeout, BFD provides rapid detection of link failures. BFD timers are adaptive and can be adjusted to be more or less aggressive. For example, a timer can adapt to a higher value if the adjacency fails, or a neighbor can negotiate a higher value for a timer than the one configured. Note that the functionality of configuring BFD for RIP described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

NOTE: EX4600 switches do not support minimum interval values of less than 1 second.

BFD enables quick failover between a primary and a secondary routed path. The protocol tests the operational status of the interface multiple times per second. BFD provides for configuration timers and thresholds for failure detection. For example, if the minimum interval is set for 50 milliseconds and the threshold uses the default value of three missed messages, a failure is detected on an interface within 200 milliseconds of the failure.

Intervening devices (for example, an Ethernet LAN switch) hide link-layer failures from routing protocol peers, such as when two routers are connected by way of a LAN switch, where the local interface status remains up even when a physical fault happens on the remote link. Link-layer failure detection times vary, depending on the physical media and the Layer 2 encapsulation. BFD can provide fast failure detection times for all media types, encapsulations, topologies, and routing protocols.

To enable BFD for RIP, both sides of the connection must receive an update message from the peer. By default, RIP does not export any routes. Therefore, you must enable update messages to be sent by configuring an export policy for routes before a BFD session is triggered.

Example: Configuring BFD for RIP

IN THIS SECTION

This example shows how to configure Bidirectional Forwarding Detection (BFD) for a RIP network.

Requirements

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

Overview

IN THIS SECTION

To enable failure detection, include the `bfd-liveness-detection` statement:

```
bfd-liveness-detection {
    detection-time {
        threshold milliseconds;
    }
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    multiplier number;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    version (1 | automatic);
}
```

Optionally, you can specify the threshold for the adaptation of the detection time by including the `threshold` statement. When the BFD session detection time adapts to a value equal to or greater than the threshold, a single trap and a system log message are sent.

To specify the minimum transmit and receive interval for failure detection, include the `minimum-interval` statement. This value represents the minimum interval at which the local routing device transmits hello packets as well as the minimum interval at which the routing device expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,000 milliseconds. This examples sets a minimum interval of 600 milliseconds.

NOTE: BFD is an intensive protocol that consumes system resources. Specifying a minimum interval for BFD of less than 100 ms for Routing Engine-based sessions and 10 ms for distributed BFD sessions can cause undesired BFD flapping.

Depending on your network environment, these additional recommendations might apply:

- For large-scale network deployments with a large number of BFD sessions, specify a minimum interval of 300 ms for Routing Engine-based sessions and 100 ms for distributed BFD sessions.

- For very large-scale network deployments with a large number of BFD sessions, contact Juniper Networks customer support for more information.

- For BFD sessions to remain up during a Routing Engine switchover event when nonstop active routing (NSR) is configured, specify a minimum interval of 2500 ms for Routing Engine-based sessions. For distributed BFD sessions with nonstop active routing configured, the minimum interval recommendations are unchanged and depend only on your network deployment.

You can optionally specify the minimum transmit and receive intervals separately.

To specify only the minimum receive interval for failure detection, include the `minimum-receive-interval` statement. This value represents the minimum interval at which the local routing device expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,00 milliseconds.

To specify only the minimum transmit interval for failure detection, include the `transmit-interval minimum-interval` statement. This value represents the minimum interval at which the local routing device transmits hello packets to the neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,000 milliseconds.

To specify the number of hello packets not received by a neighbor that causes the originating interface to be declared down, include the `multiplier` statement. The default is 3, and you can configure a value in the range from 1 through 255.

To specify the threshold for detecting the adaptation of the transmit interval, include the `transmit-interval threshold` statement. The threshold value must be greater than the transmit interval.

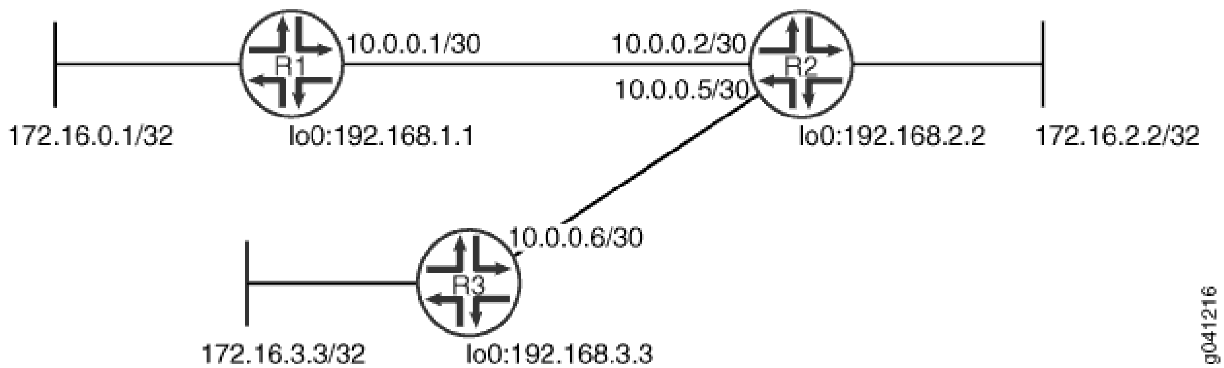
To specify the BFD version used for detection, include the `version` statement. The default is to have the version detected automatically.

You can trace BFD operations by including the `traceoptions` statement at the `[edit protocols bfd]` hierarchy level.

In Junos OS Release 9.0 and later, you can configure BFD sessions not to adapt to changing network conditions. To disable BFD adaptation, include the `no-adaptation` statement. We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

Figure 9 on page 66 shows the topology used in this example.

Figure 9: RIP BFD Network Topology



"CLI Quick Configuration" on page 66 shows the configuration for all of the devices in Figure 9 on page 66. The section "Step-by-Step Procedure" on page 67 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set protocols bfd traceoptions file bfd-trace
set protocols bfd traceoptions flag all
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.1
```

```
set protocols rip group rip-group bfd-liveness-detection minimum-interval 600
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 5 family inet address 10.0.0.5/30
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.2
set protocols rip group rip-group neighbor fe-1/2/1.5
set protocols rip group rip-group bfd-liveness-detection minimum-interval 600
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.6
set protocols rip group rip-group bfd-liveness-detection minimum-interval 600
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

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 a BFD for a RIP network:

1. Configure the network interfaces.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 family inet address 10.0.0.1/30
   ```

2. Create the RIP group and add the interface.

   To configure RIP in Junos OS, you must configure a group that contains the interfaces on which RIP is enabled. You do not need to enable RIP on the loopback interface.

   ```
   [edit protocols rip group rip-group]
   user@R1# set neighbor fe-1/2/0.1
   ```

3. Create the routing policy to advertise both direct and RIP-learned routes.

   ```
   [edit policy-options policy-statement advertise-routes-through-rip term 1]
   user@R1# set from protocol direct
   user@R1# set from protocol rip
   user@R1# set then accept
   ```

4. Apply the routing policy.

   In Junos OS, you can only apply RIP export policies at the group level.

   ```
   [edit protocols rip group rip-group]
   user@R1# set export advertise-routes-through-rip
   ```

5. Enable BFD.

   ```
   [edit protocols rip group rip-group]
   user@R1# set bfd-liveness-detection minimum-interval 600
   ```

6. Configure tracing operations to track BFD messages.

   ```
   [edit protocols bfd traceoptions]
   user@R1# set file bfd-trace
   user@R1# set flag all
   ```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        family inet {
            address 10.0.0.1/30;
        }
    }
}
```

```
user@R1# show protocols
bfd {
    traceoptions {
        file bfd-trace;
        flag all;
    }
}
rip {
    group rip-group {
        export advertise-routes-through-rip;
        bfd-liveness-detection {
            minimum-interval 600;
        }
        neighbor fe-1/2/0.1;
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-rip {
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying That the BFD Sessions Are Up

Purpose

Make sure that the BFD sessions are operating.

Action

From operational mode, enter the `show bfd session` command.

```
user@R1> show bfd session
                                                  Detect   Transmit
Address                  State     Interface      Time     Interval  Multiplier
10.0.0.2                 Up        fe-1/2/0.1     1.800     0.600        3

1 sessions, 1 clients
Cumulative transmit rate 1.7 pps, cumulative receive rate 1.7 pps
```

Meaning

The output shows that there are no authentication failures.

Checking the BFD Trace File

Purpose

Use tracing operations to verify that BFD packets are being exchanged.

Action

From operational mode, enter the `show log` command.

```
user@R1> show log bfd-trace
Feb 16 10:26:32 PPM Trace: BFD periodic xmit to 10.0.0.2 (IFL 124, rtbl 53, single-hop port)
Feb 16 10:26:32 Received Downstream TraceMsg (24) len 86:
Feb 16 10:26:32    IfIndex (3) len 4: 0
Feb 16 10:26:32    Protocol (1) len 1: BFD
Feb 16 10:26:32    Data (9) len 61: (hex) 42 46 44 20 70 61 63 6b 65 74 20 66 72 6f 6d 20 31 30
2e
Feb 16 10:26:32 PPM Trace: BFD packet from 10.0.0.1 (IFL 73, rtbl 56, ttl 255) absorbed
Feb 16 10:26:32 Received Downstream TraceMsg (24) len 60:
Feb 16 10:26:32    IfIndex (3) len 4: 0
Feb 16 10:26:32    Protocol (1) len 1: BFD
Feb 16 10:26:32    Data (9) len 35: (hex) 42 46 44 20 70 65 72 69 6f 64 69 63 20 78 6d 69 74 20
6f
...
```

Meaning

The output shows the normal functioning of BFD.

Understanding BFD Authentication for RIP

IN THIS SECTION

BFD enables rapid detection of communication failures between adjacent systems. By default, authentication for BFD sessions is disabled. However, when running BFD over Network Layer protocols, the risk of service attacks can be significant. We strongly recommend using authentication if you are running BFD over multiple hops or through insecure tunnels. Beginning with Junos OS Release 9.6, Junos OS supports authentication for BFD sessions running over RIP. BFD authentication is only supported in the domestic image and is not available in the export image.

You authenticate BFD sessions by specifying an authentication algorithm and keychain, and then associating that configuration information with a security authentication keychain using the keychain name.

The following sections describe the supported authentication algorithms, security keychains, and the level of authentication that can be configured:

BFD Authentication Algorithms

Junos OS supports the following algorithms for BFD authentication:

- **simple-password**—Plain-text password. One to 16 bytes of plain text are used to authenticate the BFD session. One or more passwords can be configured. This method is the least secure and should be used only when BFD sessions are not subject to packet interception.

- **keyed-md5**—Keyed Message Digest 5 hash algorithm for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed MD5 uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than or equal to the last sequence number received. Although more secure than a simple password, this method is vulnerable to replay attacks. Increasing the rate at which the sequence number is updated can reduce this risk.

- **meticulous-keyed-md5**—Meticulous keyed Message Digest 5 hash algorithm. This method works in the same manner as keyed MD5, but the sequence number is updated with every packet. Although more secure than keyed MD5 and simple passwords, this method might take additional time to authenticate the session.

- **keyed-sha-1**—Keyed Secure Hash Algorithm I for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed SHA uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. The key is not carried within the packets. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than the last sequence number received.

- **meticulous-keyed-sha-1**—Meticulous keyed Secure Hash Algorithm I. This method works in the same manner as keyed SHA, but the sequence number is updated with every packet. Although more secure than keyed SHA and simple passwords, this method might take additional time to authenticate the session.

NOTE: *Nonstop active routing* is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algorithms might go down after a switchover.

NOTE: QFX5000 Series switches and EX4600 switches do not support minimum interval values of less than 1 second.

Security Authentication Keychains

The security authentication keychain defines the authentication attributes used for authentication key updates. When the security authentication keychain is configured and associated with a protocol through the keychain name, authentication key updates can occur without interrupting routing and signaling protocols.

The authentication keychain contains one or more keychains. Each keychain contains one or more keys. Each key holds the secret data and the time at which the key becomes valid. The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

BFD allows multiple clients per session, and each client can have its own keychain and algorithm defined. To avoid confusion, we recommend specifying only one security authentication keychain.

Strict Versus Loose Authentication

By default, strict authentication is enabled and authentication is checked at both ends of each BFD session. Optionally, to smooth migration from nonauthenticated sessions to authenticated sessions, you can configure *loose checking*. When loose checking is configured, packets are accepted without authentication being checked at each end of the session. This feature is intended for transitional periods only.

SEE ALSO

bfd-liveness-detection (Protocols RIP) | **229**

Example: Configuring BFD Authentication for RIP

IN THIS SECTION

This example shows how to configure Bidirectional Forwarding Detection (BFD) authentication for a RIP network.

Requirements

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

The devices must be running Junos OS Release 9.6 or later.

Overview

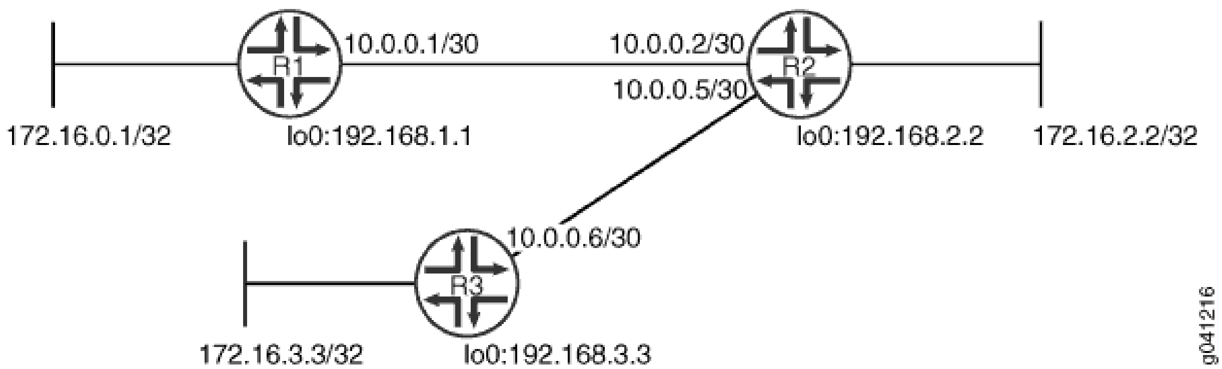
IN THIS SECTION

Only three steps are needed to configure authentication on a BFD session:

1. Specify the BFD authentication algorithm for the RIP protocol.
2. Associate the authentication keychain with the RIP protocol.
3. Configure the related security authentication keychain.

Figure 10 on page 75 shows the topology used in this example.

Figure 10: RIP BFD Authentication Network Topology



"CLI Quick Configuration" on page 75 shows the configuration for all of the devices in Figure 10 on page 75. The section "No Link Title" on page 77 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set protocols bfd traceoptions file bfd-trace
set protocols bfd traceoptions flag all
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.1
```

```
set protocols rip group rip-group bfd-liveness-detection minimum-interval 600
set protocols rip group rip-group bfd-liveness-detection authentication key-chain bfd-rip
set protocols rip group rip-group bfd-liveness-detection authentication algorithm keyed-md5
set protocols rip group rip-group bfd-liveness-detection authentication loose-check
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
set security authentication-key-chains key-chain bfd-rip key 53 secret $ABC123$ABC123
set security authentication-key-chains key-chain bfd-rip key 53 start-time "2012-2-16.12:00:00
-0800"
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 5 family inet address 10.0.0.5/30
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.2
set protocols rip group rip-group neighbor fe-1/2/1.5
set protocols rip group rip-group bfd-liveness-detection minimum-interval 600
set protocols rip group rip-group bfd-liveness-detection authentication key-chain bfd-rip
set protocols rip group rip-group bfd-liveness-detection authentication algorithm keyed-md5
set protocols rip group rip-group bfd-liveness-detection authentication loose-check
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
set security authentication-key-chains key-chain bfd-rip key 53 secret $ABC123$ABC123
set security authentication-key-chains key-chain bfd-rip key 53 start-time "2012-2-16.12:00:00
-0800"
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.6
set protocols rip group rip-group bfd-liveness-detection minimum-interval 600
set protocols rip group rip-group bfd-liveness-detection authentication key-chain bfd-rip
set protocols rip group rip-group bfd-liveness-detection authentication algorithm keyed-md5
set protocols rip group rip-group bfd-liveness-detection authentication loose-check
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

```
set security authentication-key-chains key-chain bfd-rip key 53 secret $ABC123$ABC123
set security authentication-key-chains key-chain bfd-rip key 53 start-time "2012-2-16.12:00:00
-0800"
```

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 a BFD authentication:

1. Configure the network interfaces.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 family inet address 10.0.0.1/30
   ```

2. Create the RIP group and add the interface.

   To configure RIP in Junos OS, you must configure a group that contains the interfaces on which RIP is enabled. You do not need to enable RIP on the loopback interface.

   ```
   [edit protocols rip group rip-group]
   user@R1# set neighbor fe-1/2/0.1
   ```

3. Create the routing policy to advertise both direct and RIP-learned routes.

   ```
   [edit policy-options policy-statement advertise-routes-through-rip term 1]
   user@R1# set from protocol direct
   user@R1# set from protocol rip
   user@R1# set then accept
   ```

4. Apply the routing policy.

   In Junos OS, you can only apply RIP export policies at the group level.

   ```
   [edit protocols rip group rip-group]
   user@R1# set export advertise-routes-through-rip
   ```

5. Enable BFD.

```
[edit protocols rip group rip-group]
user@R1# set bfd-liveness-detection minimum-interval 600
```

6. Specify the algorithm (**keyed-md5**, **keyed-sha-1**, **meticulous-keyed-md5**, **meticulous-keyed-sha-1**, or **simple-password**) to use.

   NOTE: Nonstop active routing is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algorithms might go down after a switchover.

```
[edit protocols rip group rip-group]
user@R1# set bfd-liveness-detection authentication algorithm keyed-md5
```

7. Specify the keychain to be used to associate BFD sessions on RIP with the unique security authentication keychain attributes.

   The keychain you specify must match a keychain name configured at the `[edit security authentication key-chains]` hierarchy level.

   The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

```
[edit protocols rip group rip-group]
user@R1# set bfd-liveness-detection authentication key-chain bfd-rip
```

8. (Optional) Specify loose authentication checking if you are transitioning from nonauthenticated sessions to authenticated sessions.

```
[edit protocols rip group rip-group]
user@R1# set bfd-liveness-detection authentication loose-check
```

9. Specify the unique security authentication information for BFD sessions:

   - The matching keychain name as specified in Step "7" on page 78.

- At least one key, a unique integer between **0** and **63**. Creating multiple keys allows multiple clients to use the BFD session.
- The secret data used to allow access to the session.
- The time at which the authentication key becomes active, in the format *yyyy-mm-dd.hh:mm:ss*.

```
[edit security authentication-key-chains key-chain bfd-rip]
user@R1# set key 53 secret $ABC123$ABC123
user@R1# set key 53 start-time "2012-2-16.12:00:00 -0800"
```

10. Configure tracing operations to track BFD authentication.

```
[edit protocols bfd traceoptions]
user@R1# set file bfd-trace
user@R1# set flag all
```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        family inet {
            address 10.0.0.1/30;
        }
    }
}
```

```
user@R1# show protocols
bfd {
    traceoptions {
        file bfd-trace;
        flag all;
    }
}
```

```
rip {
    group rip-group {
        export advertise-routes-through-rip;
        bfd-liveness-detection {
            minimum-interval 600;
        }
        neighbor fe-1/2/0.1;
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-rip {
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
```

```
user@R1# show security
authentication-key-chains {
    key-chain bfd-rip {
        key 53 {
            secret $ABC123$ABC123
            start-time "2012-2-16.12:00:00 -0800";
        }
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying That the BFD Sessions Are Authenticated

Purpose

Make sure that the BFD sessions are authenticated.

Action

From operational mode, enter the `show bfd session detail` command.

```
user@R1> show bfd session detail
                                                    Detect   Transmit
Address                   State     Interface      Time     Interval  Multiplier
10.0.0.2                  Up        fe-1/2/0.1     1.800     0.600        3
 Client RIP, TX interval 0.600, RX interval 0.600, Authenticate
 Session up time 01:39:34
 Local diagnostic None, remote diagnostic None
 Remote state Up, version 1
 Logical system 6, routing table index 53

1 sessions, 1 clients
Cumulative transmit rate 1.7 pps, cumulative receive rate 1.7 pps
```

Meaning

Authenticate is displayed to indicate that BFD authentication is configured.

Viewing Extensive Information About the BFD Authentication

Purpose

View the keychain name, the authentication algorithm and mode for each client in the session, and the BFD authentication configuration status.

Action

From operational mode, enter the `show bfd session extensive` command.

```
user@R1> show bfd session extensive
                                                  Detect   Transmit
Address                  State     Interface      Time     Interval  Multiplier
10.0.0.2                 Up        fe-1/2/0.1     1.800     0.600        3
 Client RIP, TX interval 0.600, RX interval 0.600, Authenticate
        keychain bfd-rip, algo keyed-md5, mode loose
 Session up time 01:46:29
 Local diagnostic None, remote diagnostic None
 Remote state Up, version 1
 Logical system 6, routing table index 53
 Min async interval 0.600, min slow interval 1.000
 Adaptive async TX interval 0.600, RX interval 0.600
 Local min TX interval 0.600, minimum RX interval 0.600, multiplier 3
 Remote min TX interval 0.600, min RX interval 0.600, multiplier 3
 Local discriminator 225, remote discriminator 226
 Echo mode disabled/inactive
 Authentication enabled/active, keychain bfd-rip, algo keyed-md5, mode loose
  Session ID: 0x300501

1 sessions, 1 clients
Cumulative transmit rate 1.7 pps, cumulative receive rate 1.7 pps
```

Meaning

The output shows the keychain name, the authentication algorithm and mode for the client in the session, and the BFD authentication configuration status.

Checking the BFD Trace File

Purpose

Use tracing operations to verify that BFD packets are being exchanged.

Action

From operational mode, enter the `show log` command.

```
user@R1> show log bfd-trace
Feb 16 10:26:32 PPM Trace: BFD periodic xmit to 10.0.0.2 (IFL 124, rtbl 53, single-hop port)
Feb 16 10:26:32 Received Downstream TraceMsg (24) len 86:
Feb 16 10:26:32    IfIndex (3) len 4: 0
Feb 16 10:26:32    Protocol (1) len 1: BFD
Feb 16 10:26:32    Data (9) len 61: (hex) 42 46 44 20 70 61 63 6b 65 74 20 66 72 6f 6d 20 31 30
2e
Feb 16 10:26:32 PPM Trace: BFD packet from 10.0.0.1 (IFL 73, rtbl 56, ttl 255) absorbed
Feb 16 10:26:32 Received Downstream TraceMsg (24) len 60:
Feb 16 10:26:32    IfIndex (3) len 4: 0
Feb 16 10:26:32    Protocol (1) len 1: BFD
Feb 16 10:26:32    Data (9) len 35: (hex) 42 46 44 20 70 65 72 69 6f 64 69 63 20 78 6d 69 74 20
6f
...
```

Meaning

The output shows the normal functioning of BFD.

Release History Table

Release	Description
15.1X49	Note that the functionality of configuring BFD for RIP described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

Traffic Control in a RIP Network

IN THIS SECTION

Understanding Traffic Control with Metrics in a RIP Network

To tune a RIP network and control traffic flowing through the network, you increase or decrease the cost of the paths through the network. RIP provides two ways to modify the path cost: an incoming metric and an outgoing metric, which are each set to **1** by default. These metrics are attributes that manually specify the cost of any route advertised through a host. By increasing or decreasing the metrics—and thus the cost—of links throughout the network, you can control packet transmission across the network.

The incoming metric modifies the cost of an individual segment when a route across the segment is imported into the routing table. For example, if you set the incoming metric on the segment to **3**, the individual segment cost along the link is changed from **1** to **3**. The increased cost affects all route calculations through that link. Other routes that were previously excluded because of a high hop count might now be selected into the router's forwarding table.

The outgoing metric modifies the path cost for all the routes advertised out a particular interface. Unlike the incoming metric, the outgoing metric modifies the routes that other routers are learning and thereby controls the way they send traffic.

If an exported route was learned from a member of the same RIP group, the metric associated with that route is the normal RIP metric. For example, a RIP route with a metric of 5 learned from a neighbor configured with an incoming metric of 2 is advertised with a combined metric of 7 when advertised to neighbors in the same group. However, if this route was learned from a RIP neighbor in a different group or from a different protocol, the route is advertised with the metric value configured in the outgoing metric for that group.

You might want to increase the metric of routes to decrease the likelihood that a particular route is selected and installed in the routing table. This process is sometimes referred to as *route poisoning*. Some reasons that you might want to poison a route are that the route is relatively expensive to use, or it has relatively low bandwidth.

A route with a higher metric than another route becomes the active route only when the lower-metric route becomes unavailable. In this way, the higher-metric route serves as a backup path.

One way to increase the metric of imported routes is to configure an import policy. Another way is to include the `metric-in` statement in the RIP neighbor configuration. One way to increase the metric of export routes is to configure an export policy. Another way is to include the `metric-out` statement in the RIP neighbor configuration.

Example: Controlling Traffic in a RIP Network with an Incoming Metric

IN THIS SECTION

This example shows how to control traffic with an incoming metric.

Requirements

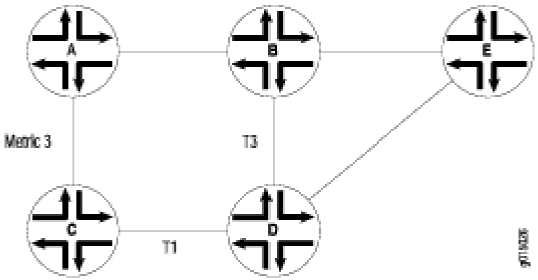
Before you begin, define RIP groups, and add interfaces to the groups. Then configure a routing policy to export directly connected routes and routes learned through the RIP routing exchanges. See "Example: Configuring a Basic RIP Network" on page 13.

Overview

IN THIS SECTION

In this example, routes to Router D are received by Router A across both of its RIP-enabled interfaces as shown in Figure 11 on page 86. Because the route through Router B and the route through Router C have the same number of hops, both routes are imported into the forwarding table. However, because the T3 link from Router B to Router D has a higher bandwidth than the T1 link from Router C to Router D, you want traffic to flow from Router A through Router B to Router D.

Figure 11: Controlling Traffic in a RIP Network with the Incoming Metric



Topology

To force this flow, you can modify the route metrics as they are imported into Router A's routing table. By setting the incoming metric on the interface from Router A to Router C, you modify the metric on all routes received through that interface. Setting the incoming route metric on Router A changes only the routes in Router A's routing table, and affects only how Router A sends traffic to Router D. Router D's route selection is based on its own routing table, which, by default, includes no adjusted metric values.

In the example, Router C receives a route advertisement from Router D and readvertises the route to Router A. When Router A receives the route, it applies the incoming metric on the interface. Instead of incrementing the metric by 1 (the default), Router A increments it by 3 (the configured incoming metric), giving the route from Router A to Router D through Router C a total path metric of 4. Because the route through Router B has a metric of 2, it becomes the preferred route for all traffic from Router A to Router D.

This example uses a RIP group called **alpha 1** on interface **g3–0/0/0**.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To control traffic with an incoming metric:

1. Enable RIP on the interface.

```
[edit protocols rip]
user@host# set group alpha1 neighbor ge-0/0/0
```

2. Set the incoming metric.

```
[edit protocols rip]
user@host# set metric-in 3
```

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

```
[edit]
user@host# commit
```

Verification

To verify that the configuration is working properly, enter the `show route protocols rip` command.

Example: Controlling Traffic in a RIP Network with an Outgoing Metric

IN THIS SECTION

This example shows how to control traffic with an outgoing metric.

Requirements

Before you begin:

- Define RIP groups, and add interfaces to the groups. Then configure a routing policy to export directly connected routes and routes learned through RIP routing exchanges. See "Example: Configuring a Basic RIP Network" on page 13.

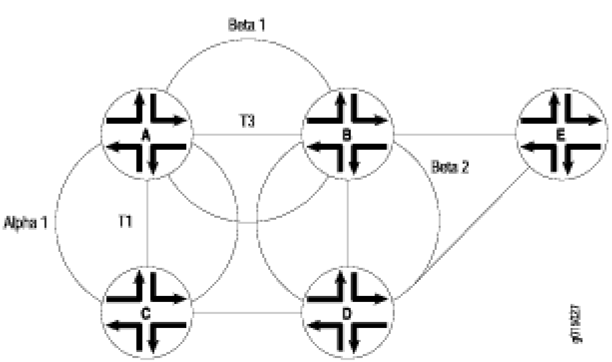
- Control traffic with an incoming metric. See "Example: Controlling Traffic in a RIP Network with an Incoming Metric" on page 85.

Overview

IN THIS SECTION

In this example, each route from Router A to Router D has two hops as shown in Figure 12 on page 88. However, because the link from Router A to Router B in the RIP group has a higher bandwidth than the link from Router A to Router C in RIP group Alpha 1, you want traffic from Router D to Router A to flow through Router B. To control the way Router D sends traffic to Router A, you can alter the routes that Router D receives by configuring the outgoing metric on Router A's interfaces in the Alpha 1 RIP group.

Figure 12: Controlling Traffic in a RIP Network with the Outgoing Metric



Topology

If the outgoing metric for the Alpha 1 RIP group—the A-to-C link—is changed to 3, Router D calculates the total path metric from Router A through Router C as 4. In contrast, the unchanged default total path metric to Router A through Router B in the RIP group is 2. The fact that Router A's interfaces belong to two different RIP groups allows you to configure two different outgoing metrics on its interfaces, because you configure path metrics at the group level.

By configuring the outgoing metric, you control the way Router A sends traffic to Router D. By configuring the outgoing metric on the same router, you control the way Router D sends traffic to Router A.

This example uses an outgoing metric of **3**.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To control traffic with an outgoing metric:

1. Set the outgoing metric.

```
[edit protocols rip group alpha1]
user@host# set metric-out 3
```

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

```
[edit]
user@host# commit
```

Verification

To verify that the configuration is working properly, enter the `show protocols rip` command.

Example: Configuring the Metric Value Added to Imported RIP Routes

IN THIS SECTION

This example shows how to change the default metric to be added to incoming routes to control the route selection process.

Requirements

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

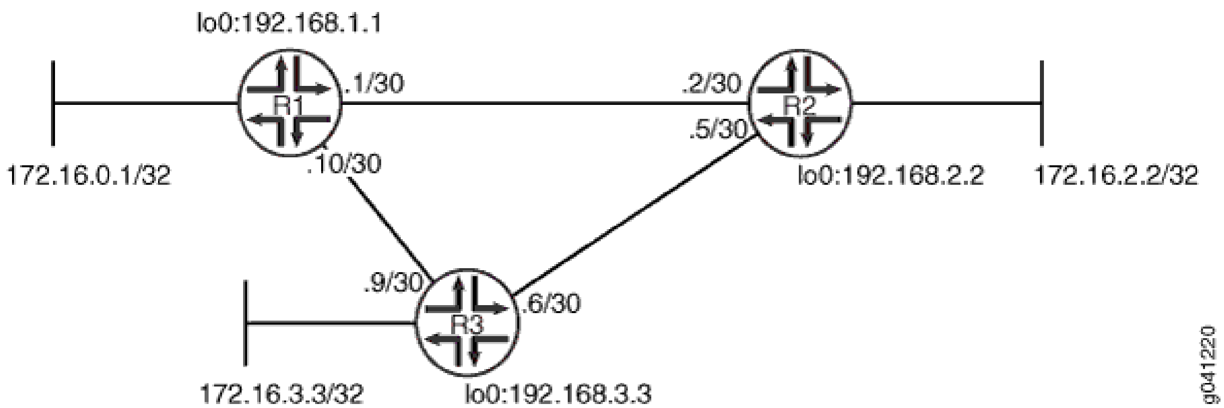
Overview

IN THIS SECTION

Normally, when multiple routes are available, RIP selects the route with the lowest hop count. Changing the default metric enables you to control the route selection process such that a route with a higher hop count can be preferred over of a route with a lower hop count.

Figure 13 on page 91 shows the topology used in this example.

Figure 13: RIP Incoming Metrics Network Topology



Device R1 has two potential paths to reach 172.16.2.2/32. The default behavior is to send traffic out the 0.1/30 interface facing Device R2. Suppose, though, that the path through Device R3 is less expensive to use or has higher bandwidth links. This example shows how to use the `metric-in` statement to ensure that Device R1 uses the path through Device R3 to reach 172.16.2.2/32. "CLI Quick Configuration" on page 91 shows the configuration for all of the devices in Figure 13 on page 91. The section "No Link Title" on page 93 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 description to-R2
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces ge-1/2/1 unit 10 description to-R3
set interfaces ge-1/2/1 unit 10 family inet address 10.0.0.10/30
set interfaces lo0 unit 1 family inet address 172.16.0.1/32
set interfaces lo0 unit 1 family inet address 192.168.1.1/32
set protocols rip group primary export advertise-routes-through-rip
set protocols rip group primary neighbor ge-1/2/1.10
set protocols rip group secondary export advertise-routes-through-rip
set protocols rip group secondary neighbor fe-1/2/0.1 metric-in 4
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces ge-1/2/1 unit 5 family inet address 10.0.0.5/30
set interfaces lo0 unit 2 family inet address 192.168.2.2/32
set interfaces lo0 unit 2 family inet address 172.16.2.2/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.2
set protocols rip group rip-group neighbor ge-1/2/1.5
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set interfaces ge-1/2/1 unit 9 family inet address 10.0.0.9/30
set interfaces lo0 unit 3 family inet address 192.168.3.3/32
set interfaces lo0 unit 3 family inet address 172.16.3.3/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.6
set protocols rip group rip-group neighbor ge-1/2/1.9
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
```

```
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

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 a RIP metrics:

1. Configure the network interfaces.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 description to-R2
   user@R1# set fe-1/2/0 unit 1 family inet address 10.0.0.1/30
   user@R1# set ge-1/2/1 unit 10 description to-R3
   user@R1# set ge-1/2/1 unit 10 family inet address 10.0.0.10/30
   user@R1# set lo0 unit 1 family inet address 172.16.0.1/32
   user@R1# set lo0 unit 1 family inet address 192.168.1.1/32
   ```

2. Create the RIP groups and add the interfaces.

   To configure RIP in Junos OS, you must configure one or more groups that contain the interfaces on which RIP is enabled. You do not need to enable RIP on the loopback interface.

   For the interface that is facing Device R2, the **metric-in 4** setting causes this route to be less likely to be chosen as the active route.

   ```
   [edit protocols rip]
   user@R1# set group primary neighbor ge-1/2/1.10
   user@R1# set group secondary neighbor fe-1/2/0.1 metric-in 4
   ```

3. Create the routing policy to advertise both direct and RIP-learned routes.

   ```
   [edit policy-options policy-statement advertise-routes-through-rip term 1]
   user@R1# set from protocol direct
   user@R1# set from protocol rip
   user@R1# set then accept
   ```

4. Apply the routing policy.

   In Junos OS, you can only apply RIP export policies at the group level.

   ```
   [edit protocols rip]
   user@R1# set group primary export advertise-routes-through-rip
   user@R1# set group secondary export advertise-routes-through-rip
   ```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        description to-R2;
        family inet {
            address 10.0.0.1/30;
        }
    }
}
ge-1/2/1 {
    unit 10 {
        description to-R3;
        family inet {
            address 10.0.0.10/30;
        }
    }
}
lo0 {
    unit 1 {
        family inet {
            address 172.16.0.1/32;
            address 192.168.1.1/32;
        }
```

```
    }
}
```

```
user@R1# show protocols
rip {
    group primary {
        export advertise-routes-through-rip;
        neighbor ge-1/2/1.10;
    }
    group secondary {
        export advertise-routes-through-rip;
        neighbor fe-1/2/0.1 {
            metric-in 4;
        }
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-rip {
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying That the Expected Route Is Active

Purpose

Make sure that to reach 172.16.2.2/32, Device R1 uses the path through Device R3.

Action

From operational mode, enter the `show route 172.16.2.2` command.

```
user@R1> show route 172.16.2.2
inet.0: 12 destinations, 12 routes (12 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.2.2/32      *[RIP/100] 00:15:46, metric 3, tag 0
                    > to 10.0.0.9 via ge-1/2/1.10
```

Meaning

The **to 10.0.0.9 via ge-1/2/1.10** output shows that Device R1 uses the path through Device R3 to reach 172.16.2.2/32. The metric for this route is 3.

Removing the metric-in Statement

Purpose

Delete or deactivate the `metric-in` statement to see what happens to the 172.16.2.2/32 route.

Action

1. From configuration mode, deactivate the `metric-in` statement.

   ```
   [edit protocols rip group secondary neighbor fe-1/2/0.1]
   user@R1# deactivate metric-in
   user@R1# commit
   ```

2. From operational mode, enter the `show route 172.16.2.2` command.

```
user@R1> show route 172.16.2.2
inet.0: 12 destinations, 12 routes (12 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.2.2/32      *[RIP/100] 00:00:06, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
```

Meaning

The **to 10.0.0.2 via fe-1/2/0.1** output shows that Device R1 uses the path through Device R2 to reach 172.16.2.2/32. The metric for this route is 2.

RELATED DOCUMENTATION

RIP Configuration Overview | **12**

Verifying a RIP Configuration | **216**

RIP Overview | **2**

Point-to-Multipoint RIP Networks

IN THIS SECTION

Configuring Point-to-Multipoint RIP Networks Overview

A point-to-multipoint RIP network consists of a device having two or more peers on a single interface. All the devices forming a point-to-multipoint connection are placed in a single broadcast domain.

In a RIP network, a device can have a single peer or multiple peers for an interface. However, the demand circuit feature implementation in a RIP network requires the use of a single RIP peer. When you configure the following statements, a RIP network with demand circuits can also be configured to have multiple peers on an interface:

- Configuring the interface type to be a multipoint interface by using the `interface-type (Protocols RIP) p2mp` statement.
- Enabling dynamic peer discovery by using the `dynamic-peers` statement (SRX Series Firewalls only).

> **NOTE**: Before configuring the `dynamic-peers` statement, IPsec must be configured and IPsec tunnels must be set up by configuring IPsec parameters. Without IPsec configuration, the remote peers have to be explicitly configured at the RIP protocol level by using the `peer` *address* statement. See Configuring Security Associations for IPsec on an ES PIC for more details.

- Configuring peers by using the `peer` *address* statement.

```
[edit]
protocols {
    rip {
        group red {
            neighbor fe-0/1/3 {
                interface-type (Protocols RIP) p2mp;
                peer address; (or use dynamic-peers;)
            }
        }
    }
}
```

The `show rip statistics peer` *address* command can be used to display the RIP statistics at the peer level. The `clear rip statistics peer` *address* command can be used to clear the RIP statistics for a peer. Alternatively, you can use the **show rip statistics peer all** and `clear rip statistics peer all` command to display and clear RIP statistics for all peers.

Example: Configuring Point-to-Multipoint RIP Networks

IN THIS SECTION

This example shows how to configure a point-to-multipoint RIP network.

Requirements

This example uses the following hardware and software components:

- M Series routers, MX Series routers, T Series routers, or SRX Series Firewalls
- Junos OS Release 12.1 or later

Overview

IN THIS SECTION

In a RIP network, a device can have a single peer or multiple peers for an interface. However, the demand circuit feature implementation in a RIP network requires the use of a single RIP peer.

When you include the following statements, the demand circuit implementation can have multiple peers for a given RIP neighbor.

- Configuring the interface type to be a multipoint interface by using the `interface-type (Protocols RIP) p2mp` statement.
- Enabling dynamic peer discovery by using the "dynamic-peers" on page 236 statement (SRX Series Firewalls only).

NOTE: To configure the `dynamic-peers` statement, IPsec tunnels must be set up by configuring IPsec parameters. See Configuring Security Associations for IPsec on an ES PIC for more details.

- Configuring peers by using the `peer` *address* statement.

```
[edit]
protocols {
    rip {
        group red {
            neighbor fe-0/1/3 {
                interface-type (Protocols RIP) p2mp;
                peer address; (or use dynamic-peers;)
            }
        }
    }
}
```

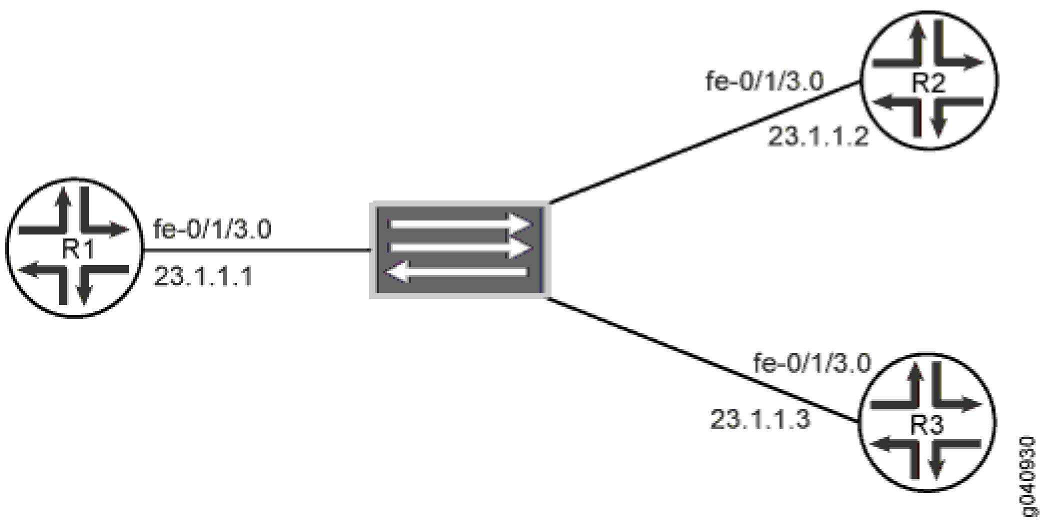
The `show rip statistics peer` command can be used to display the RIP statistics at the peer level.

Topology

In this example, Devices R1, R2, and R3 form a point-to-multipoint network. R1 is connected to R2 and to R3 as a point-to-multipoint connection through a switch that places all devices in the same broadcast domain. RIP demand circuits are configured on all three devices. The two peers to R1 are configured statically by using the `peer` *address* statement. The `dynamic-peers` statement is not used here.

Figure 14 on page 101 shows the topology used in this example.

Figure 14: Configuring a Point-to-Multipoint RIP Network



Configuration

IN THIS SECTION

CLI Quick Configuration

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

Device R1

```
set interfaces fe-0/1/3 unit 0 family inet address 23.1.1.1/24
set policy-options policy-statement accept-rip-routes term from-direct from protocol direct
set policy-options policy-statement accept-rip-routes term from-direct then accept
set policy-options policy-statement accept-rip-routes term from-rip from protocol rip
set policy-options policy-statement accept-rip-routes term from-rip then accept
set protocols rip traceoptions file R1.log size 4m world-readable
set protocols rip traceoptions flag all detail
set protocols rip group red export accept-rip-routes
```

```
set protocols rip group red neighbor fe-0/1/3.0 interface-type p2mp
set protocols rip group red neighbor fe-0/1/3.0 peer 23.1.1.2
set protocols rip group red neighbor fe-0/1/3.0 peer 23.1.1.3
set protocols rip group red neighbor fe-0/1/3.0 demand-circuit
set protocols rip group red neighbor fe-0/1/3.0 max-retrans-time 10
```

Similarly, configure Devices R2 and R3, omitting the **peer *address*** configuration statement.

Configuring a Point-to-Multipoint RIP Network (with Demand Circuits)

Step-by-Step Procedure

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

To configure the point-to-multipoint feature across a RIP network:

1. Configure the device interface.

```
[edit interfaces fe-0/1/3 unit 0]
user@R1# set family inet address 23.1.1.1/24
```

2. Define a policy for exporting RIP routes from the routing table to the protocol for transmission through the network.

```
[edit policy-options policy-statement accept-rip-routes]
user@R1# set term from-direct from protocol direct
user@R1# set term from-direct then accept
user@R1# set term from-rip from protocol rip
user@R1# set term from-rip then accept
```

3. Configure RIP and a RIP group with the defined export policy and point-to-multipoint configuration statements.

```
[edit protocols rip]
user@R1# set traceoptions file R1.log size 4m world-readable
user@R1# set traceoptions flag all detail
user@R1# set group red export accept-rip-routes
user@R1# set group red neighbor fe-0/1/3.0 interface-type p2mp
```

```
user@R1# set group red neighbor fe-0/1/3.0 peer 23.1.1.2
user@R1# set group red neighbor fe-0/1/3.0 peer 23.1.1.3
user@R1# set group red neighbor fe-0/1/3.0 demand-circuit
user@R1# set group red neighbor fe-0/1/3.0 max-retrans-time 10
```

Similarly, configure Devices R2 and R3, omitting the **peer *address*** configuration statement.

NOTE: Configuring **max-retrans-time** is optional. In the absence of this configuration statement, the default retransmission time of 180 seconds is configured.

The configuration used in this example is for a RIP network with demand circuits. To configure RIP for networks without demand circuits, exclude the **demand-circuit** and `max-retrans-time` statements from the configuration and check the resulting output. For more information about configuring RIP demand circuits, see "Example: Configuring RIP Demand Circuits" on page 57.

Results

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

```
[edit]
user@R1# show interfaces
fe-0/1/3 {
    unit 0 {
        family inet {
            address 23.1.1.1/24;
        }
    }
}
```

```
user@R1# show protocols rip
traceoptions {
    file R1.log size 4m world-readable;
    flag all detail;
}
group red {
    export accept-rip-routes;
    neighbor fe-0/1/3.0 {
```

```
        interface-type p2mp;
        peer 23.1.1.2;
        peer 23.1.1.3;
        demand-circuit;
        max-retrans-time 10;
    }
}
```

```
user@R1# show policy-options
policy-statement accept-rip-routes {
    term from-direct {
        from protocol direct;
        then accept;
    }
    term from-rip {
        from protocol rip;
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Point-to-Multipoint RIP Network

Purpose

Verify that the RIP network is functional with the point-to-multipoint feature configured.

Action

From operational mode, run the `show rip neighbor` command.

```
user@R1> show rip neighbor

                  Local  Source          Destination     Send   Receive   In
Neighbor          State  Address         Address         Mode   Mode     Met
--------          -----  -------         -----------     ----   -------  ---
fe-0/1/3.0(DC)       Up  23.1.1.1       23.1.1.2        unicast unicast    1
fe-0/1/3.0(DC)       Up  23.1.1.1       23.1.1.3        unicast unicast    1
```

From operational mode, run the `show rip statistics peer` *address* command.

```
user@R1> show rip statistics peer 23.1.1.2

RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              3              0              0              0

fe-0/1/3.0 Peer-Ip 23.1.1.2:  2 routes learned; 3 routes advertised; timeout 180s; update
interval 0s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                        0            0            0
Triggered Updates Sent              3            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received              2            0            0
RIPv2 Bad Route Entries             0            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               0            0            0
```

```
RIP Requests Ignored                0           0           0
none                                3           0           0
```

```
user@R1> show rip statistics peer 23.1.1.3

RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              3              0              0              0

fe-0/1/3.0 Peer-Ip 23.1.1.3:  2 routes learned; 3 routes advertised; timeout 180s; update
interval 0s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                        0           0           0
Triggered Updates Sent              3           0           0
Responses Sent                      0           0           0
Bad Messages                        0           0           0
RIPv1 Updates Received              0           0           0
RIPv1 Bad Route Entries             0           0           0
RIPv1 Updates Ignored               0           0           0
RIPv2 Updates Received              2           0           0
RIPv2 Bad Route Entries             0           0           0
RIPv2 Updates Ignored               0           0           0
Authentication Failures             0           0           0
RIP Requests Received               0           0           0
RIP Requests Ignored                0           0           0
none                                3           0           0
```

Meaning

The RIP network is up and running with the point-to-multipoint feature configured.

RELATED DOCUMENTATION

dynamic-peers | **236**

Example: Configuring RIP Demand Circuits | **57**

interface-type (Protocols RIP) | **248**

peer (Protocols RIP) | **259**

RIP Import Policy

IN THIS SECTION

Understanding RIP Import Policy

The default RIP import policy is to accept all received RIP routes that pass a sanity check. To filter routes being imported by the local routing device from its neighbors, include the `import` statement, and list the names of one or more policies to be evaluated. If you specify more than one policy, they are evaluated in order (first to last) and the first matching policy is applied to the route. If no match is found, the local routing device does not import any routes. Note that the functionality of applying policies to RIP routes imported from neighbors described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

Example: Applying Policies to RIP Routes Imported from Neighbors

IN THIS SECTION

This example shows how to configure an import policy in a RIP network.

Requirements

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

Overview

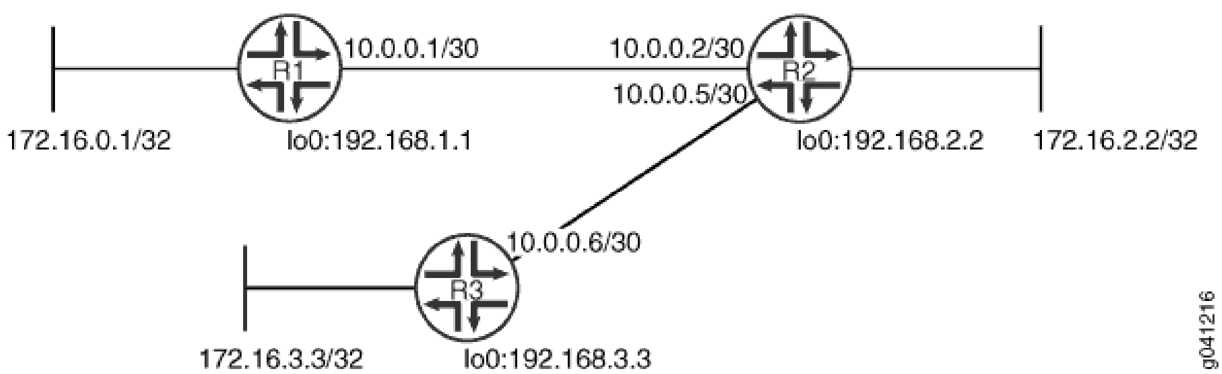
IN THIS SECTION

In this example, Device R1 has an import policy that accepts the 10/8 and 192.168/16 RIP routes and rejects all other RIP routes. This means that the 172.16/16 RIP routes are excluded from Device R1's routing table.

An export policy is also shown because an export policy is required as part of the minimum configuration for RIP.

Figure 15 on page 108 shows the topology used in this example.

Figure 15: RIP Import Policy Network Topology



"CLI Quick Configuration" on page 109 shows the configuration for all of the devices in Figure 15 on page 108. The section "No Link Title" on page 110 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces lo0 unit 1 family inet address 172.16.0.1/32
set interfaces lo0 unit 1 family inet address 192.168.1.1/32
set protocols rip import rip-import
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.1
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
set policy-options policy-statement rip-import term 1 from protocol rip
set policy-options policy-statement rip-import term 1 from route-filter 10.0.0.0/8 orlonger
set policy-options policy-statement rip-import term 1 from route-filter 192.168.0.0/16 orlonger
set policy-options policy-statement rip-import term 1 then accept
set policy-options policy-statement rip-import term 2 then reject
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 5 family inet address 10.0.0.5/30
set interfaces lo0 unit 2 family inet address 192.168.2.2/32
set interfaces lo0 unit 2 family inet address 172.16.2.2/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.2
set protocols rip group rip-group neighbor fe-1/2/1.5
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set interfaces lo0 unit 3 family inet address 192.168.3.3/32
set interfaces lo0 unit 3 family inet address 172.16.3.3/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.6
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

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 a RIP import policy:

1. Configure the network interfaces.

   This example shows multiple loopback interface addresses to simulate attached networks.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 family inet address 10.0.0.1/30
   user@R1# set lo0 unit 1 family inet address 172.16.0.1/32
   user@R1# set lo0 unit 1 family inet address 192.168.1.1/32
   ```

2. Create the RIP group and add the interface.

   To configure RIP in Junos OS, you must configure a group that contains the interfaces on which RIP is enabled.

   You do not need to enable RIP on the loopback interface.

   ```
   [edit protocols rip group rip-group]
   user@R1# set neighbor fe-1/2/0.1
   ```

3. Create the routing policy to advertise both direct and RIP-learned routes.

```
[edit policy-options policy-statement advertise-routes-through-rip term 1]
user@R1# set from protocol direct
user@R1# set from protocol rip
user@R1# set then accept
```

4. Apply the routing policy.

   In Junos OS, you can only apply RIP export policies at the group level.

```
[edit protocols rip group rip-group]
user@R1# set export advertise-routes-through-rip
```

5. Configure the import policy.

```
[edit policy-options policy-statement rip-import]
user@R1# set term 1 from protocol rip
user@R1# set term 1 from route-filter 10.0.0.0/8 orlonger
user@R1# set term 1 from route-filter 192.168.0.0/16 orlonger
user@R1# set term 1 then accept
user@R1# set term 2 then reject
```

6. Apply the import policy.

```
[edit protocols rip]
user@R1# set import rip-import
```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        family inet {
```

```
            address 10.0.0.1/30;
        }
    }
}
lo0 {
    unit 1 {
        family inet {
            address 172.16.0.1/32;
            address 192.168.1.1/32;
        }
    }
}
```

```
user@R1# show protocols
rip {
    import rip-import;
    group rip-group {
        export advertise-routes-through-rip;
        neighbor fe-1/2/0.1;
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-rip {
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
policy-statement rip-import {
    term 1 {
        from {
            protocol rip;
            route-filter 10.0.0.0/8 orlonger;
            route-filter 192.168.0.0/16 orlonger;
        }
        then accept;
    }
    term 2 {
        then reject;
```

```
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Looking at the Routes That Device R2 Is Advertising to Device R1

Purpose

Verify that Device R2 is sending the expected routes.

Action

From operational mode, enter the `show route advertising-protocol rip` command.

```
user@R2> show route advertising-protocol rip 10.0.0.2

inet.0: 11 destinations, 11 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.4/30        *[Direct/0] 2d 01:17:44
                    >    via fe-1/2/0.5
172.16.2.2/32      *[Direct/0] 2d 04:09:52
                    >    via lo0.2
172.16.3.3/32      *[RIP/100] 23:40:02, metric 2, tag 0
                    > to 10.0.0.6 via fe-1/2/0.5
192.168.2.2/32     *[Direct/0] 2d 04:09:52
```

```
                    >    via lo0.2
192.168.3.3/32     *[RIP/100] 23:40:02, metric 2, tag 0
                    > to 10.0.0.6 via fe-1/2/0.5
```

Meaning

Device R2 is sending 172.16/16 routes to Device R1.

Looking at the Routes That Device R1 Is Receiving from Device R2

Purpose

Verify that Device R1 is receiving the expected routes.

Action

From operational mode, enter the `show route receive-protocol rip` command.

```
user@R1> show route receive-protocol rip 10.0.0.2
inet.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.4/30        *[RIP/100] 01:06:03, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
192.168.2.2/32     *[RIP/100] 01:06:03, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
192.168.3.3/32     *[RIP/100] 01:06:03, metric 3, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
```

Meaning

The output shows that the 172.16/16 routes are excluded.

Checking the Routing Table

Purpose

Verify that the routing table is populated with the expected routes.

Action

From operational mode, enter the `show route protocol rip` command.

```
user@R1> show route protocol rip

inet.0: 8 destinations, 8 routes (8 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.4/30        *[RIP/100] 00:54:34, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
192.168.2.2/32     *[RIP/100] 00:54:34, metric 2, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
192.168.3.3/32     *[RIP/100] 00:54:34, metric 3, tag 0
                    > to 10.0.0.2 via fe-1/2/0.1
224.0.0.9/32       *[RIP/100] 00:49:00, metric 1
                         MultiRecv
```

Meaning

The output shows that the routes have been learned from Device R2 and Device R3.

If you delete or deactivate the import policy, the routing table contains the 172.16/16 routes.

Testing the Import Policy

Purpose

By using the `test policy` command, monitor the number of rejected prefixes.

Action

From operational mode, enter the `test policy rip-import 172.16/16` command.

```
user@R1> test policy rip-import 172.16/16
Policy rip-import: 0 prefix accepted, 1 prefix rejected
```

Meaning

The output shows that the policy rejected one prefix.

Release History Table

Release	Description
15.1X49	Note that the functionality of applying policies to RIP routes imported from neighbors described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

Interoperability of RIPv1 and RIPv2 Networks

IN THIS SECTION

Understanding the Sending and Receiving of RIPv1 and RIPv2 Packets

RIP version 1 (RIPv1) and RIP version 2 (RIPv2) can run simultaneously. This might make sense when you are migrating a RIPv1 network to a RIPv2 network. This also allows interoperation with a device that supports RIPv1 but not RIPv2.

By default, when RIP is enabled on an interface, Junos OS receives both RIPv1 and RIPv2 packets and sends only RIPv2 packets. You can configure this behavior by including the **send** and `receive` statements in the RIP configuration.Note that the functionality of configuring the sending and receiving of RIPv1 and RIPv2 packets described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

Example: Configuring the Sending and Receiving of RIPv1 and RIPv2 Packets

IN THIS SECTION

This example shows how to configure whether the RIP update messages conform to RIP version 1 (RIPv1) only, to RIP version 2 (RIPv2) only, or to both versions. You can also disable the sending or receiving of update messages.

Requirements

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

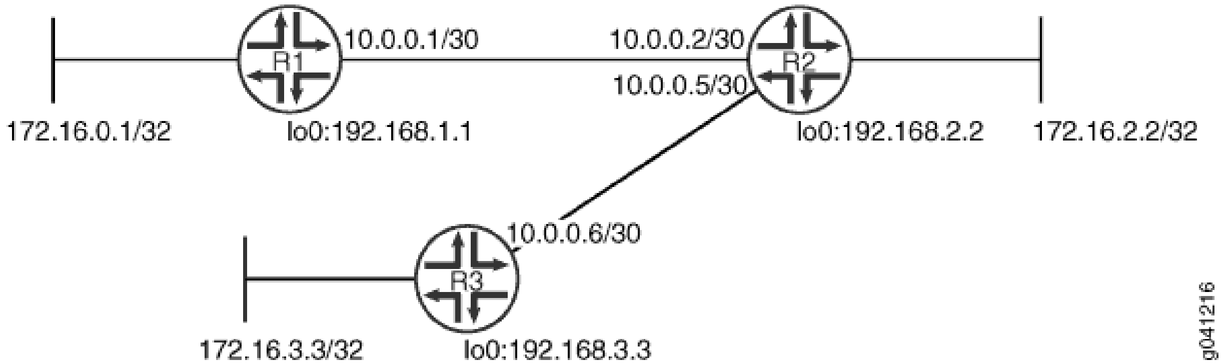
Overview

IN THIS SECTION

By default, when RIP is enabled on an interface, Junos OS receives both RIPv1 and RIPv2 packets and sends only RIPv2 packets.

Figure 16 on page 118 shows the topology used in this example.

Figure 16: Sending and Receiving RIPv1 and RIPv2 Packets Network Topology



In this example, Device R1 is configured to receive only RIPv2 packets.

"CLI Quick Configuration" on page 118 shows the configuration for all of the devices in Figure 16 on page 118. The section "No Link Title" on page 119 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces lo0 unit 1 family inet address 172.16.0.1/32
set interfaces lo0 unit 1 family inet address 192.168.1.1/32
set protocols rip group rip-group export advertise-routes-through-rip
```

```
set protocols rip group rip-group neighbor fe-1/2/0.1 receive version-2
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 5 family inet address 10.0.0.5/30
set interfaces lo0 unit 2 family inet address 192.168.2.2/32
set interfaces lo0 unit 2 family inet address 172.16.2.2/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.2
set protocols rip group rip-group neighbor fe-1/2/1.5
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set interfaces lo0 unit 3 family inet address 192.168.3.3/32
set interfaces lo0 unit 3 family inet address 172.16.3.3/32
set protocols rip group rip-group export advertise-routes-through-rip
set protocols rip group rip-group neighbor fe-1/2/0.6
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

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 a RIP packet versions that can be received:

1. Configure the network interfaces.

```
[edit interfaces]
user@R1# set fe-1/2/0 unit 1 family inet address 10.0.0.1/30
```

```
user@R1# set lo0 unit 1 family inet address 172.16.0.1/32
user@R1# set lo0 unit 1 family inet address 192.168.1.1/32
```

2. Create the RIP groups and add the interfaces.

   To configure RIP in Junos OS, you must configure one or more groups that contain the interfaces on which RIP is enabled. You do not need to enable RIP on the loopback interface.

   For the interface that is facing Device R2, the **receive version-2** setting causes this interface to accept only RIPv2 packets.

```
[edit protocols rip group rip-group]
user@R1# set neighbor fe-1/2/0.1 receive version-2
```

3. Create the routing policy to advertise both direct and RIP-learned routes.

```
[edit policy-options policy-statement advertise-routes-through-rip term 1]
user@R1# set from protocol direct
user@R1# set from protocol rip
user@R1# set then accept
```

4. Apply the routing policy.

   In Junos OS, you can only apply RIP export policies at the group level.

```
[edit protocols rip group rip-group]
user@R1# set export advertise-routes-through-rip
```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        family inet {
            address 10.0.0.1/30;
```

```
            }
        }
    }
    lo0 {
        unit 1 {
            family inet {
                address 172.16.0.1/32;
                address 192.168.1.1/32;
            }
        }
    }
```

```
user@R1# show protocols
rip {
    group rip-group {
        export advertise-routes-through-rip;
        neighbor fe-1/2/0.1 {
            receive version-2;
        }
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-rip {
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying That the Receive Mode Is Set to RIPv2 Only

Purpose

Make sure that the interfacing Device R2 is configured to receive only RIPv2 packets, instead of both RIPv1 and RIPv2 packets.Starting in Junos OS Release 19.3R1, Junos OS supports RIP version 2 (RIPv2) for both IPv4 and IPv6 packets on ACX5448 Universal Metro routers.

Action

From operational mode, enter the `show rip neighbor` command.

```
user@R1> show rip neighbor

                  Local  Source          Destination     Send   Receive   In
Neighbor          State  Address         Address         Mode   Mode      Met
--------          -----  -------         -----------     ----   -------   ---
fe-1/2/0.1           Up 10.0.0.1        224.0.0.9       mcast  v2 only    1
```

Meaning

In the output, the **Receive Mode** field displays **v2 only**. The default **Receive Mode** is **both**.

Release History Table

Release	Description
19.3R1	Starting in Junos OS Release 19.3R1, Junos OS supports RIP version 2 (RIPv2) for both IPv4 and IPv6 packets on ACX5448 Universal Metro routers.
15.149	Note that the functionality of configuring the sending and receiving of RIPv1 and RIPv2 packets described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

Route Redistribution Between RIP Instances

IN THIS SECTION

Understanding Route Redistribution Among RIP Instances

You can redistribute routes among RIP processes. Another way to say this is to export RIP routes from one RIP instance to other RIP instances.

In Junos OS, route redistribution among routing instances is accomplished by using routing table groups, also called RIB groups. Routing table groups allow you to import and export routes from a protocol within one routing table into another routing table. Note that the functionality of redistributing routes among RIP instances described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

NOTE: In contrast, the policy-based import and export functions allow you import and export routes between different protocols within the same routing table.

Consider the following partial example:

```
protocols {
    rip {
        rib-group inet-to-voice;
    }
}
routing-instances {
    voice {
        protocols {
            rip {
                rib-group voice-to-inet;
            }
        }
```

```
    }
}
routing-options {
    rib-groups {
        inet-to-voice {
            import-rib [ inet.0 voice.inet.0 ];
        }
        voice-to-inet {
            import-rib [ voice.inet.0 inet.0 ];
        }
    }
}
```

The way to read the `import-rib` statement is as follows. Take the routes from the protocol (RIP, in this case), and import them into the primary (or local) routing table and also into any other routing tables listed after this. The primary routing table is the routing table where the routing table group is being used. That would be either **inet.0** if used in the main routing instance or **voice.inet.0** if used within the routing instance. In the **inet-to-voice** routing table group, **inet.0** is listed first because this routing table group is used in the main routing instance. In the **voice-to-inet** routing table group, **voice.inet.0** is listed first because this routing table group is used in the voice routing instance.

Example: Redistributing Routes Between Two RIP Instances

IN THIS SECTION

This example shows how to configure a RIP routing instance and control the redistribution of RIP routes between the routing instance and the primary instance.

Requirements

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

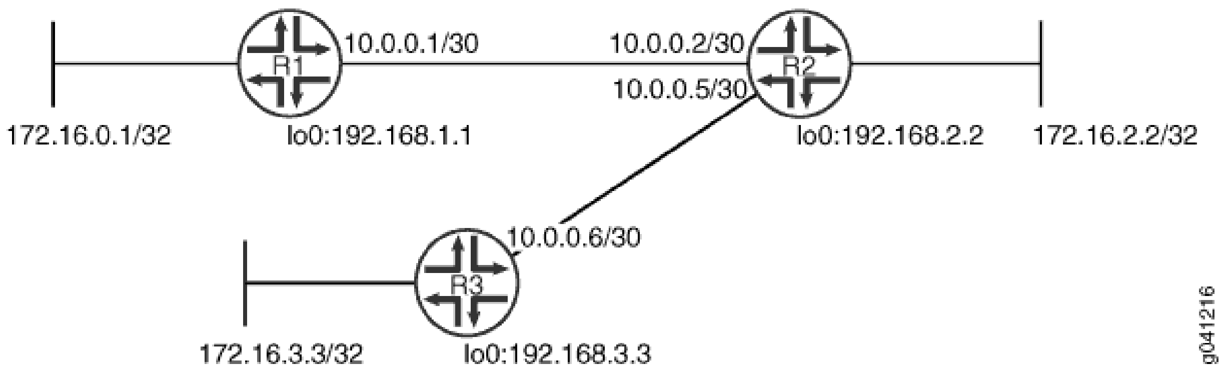
Overview

IN THIS SECTION

When you create a routing instance called voice, Junos OS creates a routing table called **voice.inet.0**. The example shows how to install routes learned through the primary RIP instance into the **voice.inet.0** routing table. The example also shows how to install routes learned through the voice routing instance into **inet.0**. This is done by configuring routing table groups. RIP routes are installed into each routing table that belongs to a routing table group.

Figure 17 on page 125 shows the topology used in this example.

Figure 17: Redistributing Routes Between RIP Instances Network Topology



"CLI Quick Configuration" on page 126 shows the configuration for all of the devices in Figure 17 on page 125. The section "No Link Title" on page 127 describes the steps on Device R2.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
set interfaces lo0 unit 1 family inet address 172.16.0.1/32
set interfaces lo0 unit 1 family inet address 192.168.1.1/32
set protocols rip group to-R2 export advertise-routes-through-rip
set protocols rip group to-R2 neighbor fe-1/2/0.1
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 5 family inet address 10.0.0.5/30
set interfaces lo0 unit 2 family inet address 192.168.2.2/32
set interfaces lo0 unit 2 family inet address 172.16.2.2/32
set protocols rip rib-group inet-to-voice
set protocols rip group to-R3 export advertise-routes-through-rip
set protocols rip group to-R3 neighbor fe-1/2/1.5
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
set routing-instances voice protocols rip group to-R1 export advertise-routes-through-rip
set routing-instances voice interface fe-1/2/0.2
set routing-instances voice protocols rip rib-group voice-to-inet
set routing-instances voice protocols rip group to-R1 neighbor fe-1/2/0.2
set routing-options rib-groups inet-to-voice import-rib inet.0
set routing-options rib-groups inet-to-voice import-rib voice.inet.0
set routing-options rib-groups voice-to-inet import-rib voice.inet.0
set routing-options rib-groups voice-to-inet import-rib inet.0
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet address 10.0.0.6/30
set interfaces lo0 unit 3 family inet address 192.168.3.3/32
set interfaces lo0 unit 3 family inet address 172.16.3.3/32
set protocols rip group to-R2 export advertise-routes-through-rip
set protocols rip group to-R2 neighbor fe-1/2/0.6
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-rip term 1 from protocol rip
set policy-options policy-statement advertise-routes-through-rip term 1 then accept
```

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 redistribute RIP routes between routing instances:

1. Configure the network interfaces.

```
[edit interfaces]
user@R2# set fe-1/2/0 unit 2 family inet address 10.0.0.2/30
user@R2# set fe-1/2/1 unit 5 family inet address 10.0.0.5/30
user@R2# set lo0 unit 2 family inet address 192.168.2.2/32
user@R2# set lo0 unit 2 family inet address 172.16.2.2/32
```

2. Create the routing instance, and add one or more interfaces to the routing instance.

```
[edit routing-instances voice]
user@R2# set interface fe-1/2/0.2
```

3. Create the RIP groups and add the interfaces.

```
[edit protocols rip group to-R3]
user@R2# set neighbor fe-1/2/1.5
[edit routing-instances voice protocols rip group to-R1]
user@R2# set neighbor fe-1/2/0.2
```

4. Create the routing table groups.

```
[edit routing-options rib-groups]
user@R2# set inet-to-voice import-rib inet.0
user@R2# set inet-to-voice import-rib voice.inet.0
user@R2# set voice-to-inet import-rib voice.inet.0
user@R2# set voice-to-inet import-rib inet.0
```

5. Apply the routing table groups.

```
[edit protocols rip]
user@R2# set rib-group inet-to-voice
[edit routing-instances voice protocols rip]
user@R2# set rib-group voice-to-inet
```

6. Create the routing policy to advertise both direct and RIP-learned routes.

```
[edit policy-options policy-statement advertise-routes-through-rip term 1]
user@R2# set from protocol direct
user@R2# set from protocol rip
user@R2# set then accept
```

7. Apply the routing policy.

   In Junos OS, you can only apply RIP export policies at the group level.

```
[edit protocols rip group to-R3]
user@R2# set export advertise-routes-through-rip
[edit routing-instances voice protocols rip group to-R1]
user@R2# set export advertise-routes-through-rip
```

Results

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

```
user@R2# show interfaces
fe-1/2/0 {
    unit 2 {
        family inet {
            address 10.0.0.2/30;
        }
    }
}
fe-1/2/1 {
    unit 5 {
        family inet {
            address 10.0.0.5/30;
        }
    }
}
lo0 {
    unit 2 {
        family inet {
            address 192.168.2.2/32;
            address 172.16.2.2/32;
        }
    }
}
```

```
user@R2# show protocols
rip {
    rib-group inet-to-voice;
    group to-R3 {
        export advertise-routes-through-rip;
        neighbor fe-1/2/1.5;
```

```
    }
}
```

```
user@R2# show policy-options
policy-statement advertise-routes-through-rip {
    term 1 {
        from protocol [ direct rip ];
        then accept;
    }
}
```

```
user@R2# show routing-instances
voice {
    interface fe-1/2/0.2;
    protocols {
        rip {
            rib-group voice-to-inet;
            group to-R1 {
                export advertise-routes-through-rip;
                neighbor fe-1/2/0.2;
            }
        }
    }
}
```

```
user@R2# show routing-options
rib-groups {
    inet-to-voice {
        import-rib [ inet.0 voice.inet.0 ];
    }
    voice-to-inet {
        import-rib [ voice.inet.0 inet.0 ];
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Checking the Routing Tables

Purpose

Make sure that the routing tables contain the expected routes.

Action

From operational mode, enter the `show route protocol rip` command.

```
user@R2> show route protocol rip
inet.0: 9 destinations, 9 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.0.1/32      *[RIP/100] 01:58:14, metric 2, tag 0
                    > to 10.0.0.1 via fe-1/2/0.2
172.16.3.3/32      *[RIP/100] 02:06:03, metric 2, tag 0
                    > to 10.0.0.6 via fe-1/2/0.5
192.168.1.1/32     *[RIP/100] 01:58:14, metric 2, tag 0
                    > to 10.0.0.1 via fe-1/2/0.2
192.168.3.3/32     *[RIP/100] 02:06:03, metric 2, tag 0
                    > to 10.0.0.6 via fe-1/2/0.5
224.0.0.9/32       *[RIP/100] 01:44:13, metric 1
                         MultiRecv

voice.inet.0: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.0.1/32      *[RIP/100] 02:06:03, metric 2, tag 0
                    > to 10.0.0.1 via fe-1/2/0.2
```

```
172.16.3.3/32      *[RIP/100] 01:58:14, metric 2, tag 0
                    > to 10.0.0.6 via fe-1/2/0.5
192.168.1.1/32     *[RIP/100] 02:06:03, metric 2, tag 0
                    > to 10.0.0.1 via fe-1/2/0.2
192.168.3.3/32     *[RIP/100] 01:58:14, metric 2, tag 0
                    > to 10.0.0.6 via fe-1/2/0.5
224.0.0.9/32       *[RIP/100] 01:44:13, metric 1
                         MultiRecv
```

Meaning

The output shows that both routing tables contain all of the RIP routes.

Release History Table

Release	Description
15.1X49	Note that the functionality of redistributing routes among RIP instances described in this topic is not supported in Junos OS Releases 15.1X49, 15.1X49-D30, or 15.1X49-D40.

3

CHAPTER

Configuring RIPng

Basic RIPng Configuration

IN THIS SECTION

Understanding Basic RIPng Routing

By default, RIP next generation (RIPng) routes are not redistributed. You must configure export policy to redistribute RIPng routes.

To have a router exchange routes with other routers, you must configure RIPng groups and neighbors. RIPng routes received from routers not configured as RIPng neighbors are ignored. Likewise, RIPng routes are advertised only to routers configured as RIPng neighbors.

Example: Configuring a Basic RIPng Network

IN THIS SECTION

This example shows how to configure a basic RIPng network.

Requirements

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

Overview

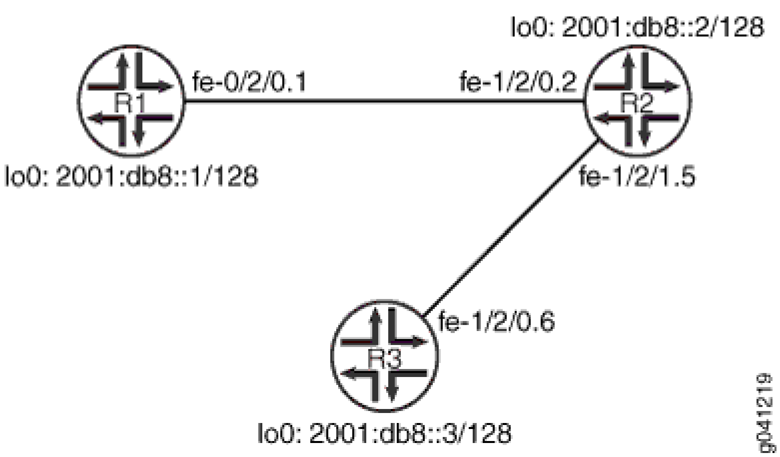
IN THIS SECTION

In this example, you configure a basic RIPng network, create a RIPng group called **ripng-group**, and add the directly connected interfaces to the RIPng group. Then you configure a routing policy to advertise direct routes using the policy statement **advertise-routes-through-ripng**.

By default, Junos OS does not advertise RIPng routes, not even routes that are learned through RIPng. To advertise RIPng routes, you must configure and apply an export routing policy that advertises RIPng-learned and direct routes.

To use RIPng on the device, you must configure RIPng on all of the RIPng interfaces within the network. Figure 18 on page 135 shows the topology used in this example.

Figure 18: Sample RIPng Network Topology



"CLI Quick Configuration" on page 136 shows the configuration for all of the devices in Figure 18 on page 135. The section "No Link Title" on page 137 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 description to-R2
set interfaces fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces lo0 unit 1 family inet6 address 2001:db8::1/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.1
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 description to-R1
set interfaces fe-1/2/0 unit 2 family inet6 address 2001:db8:0:2::/64 eui-64
set interfaces fe-1/2/1 unit 5 description to-R3
set interfaces fe-1/2/1 unit 5 family inet6 address 2001:db8:0:3::/64 eui-64
set interfaces lo0 unit 2 family inet6 address 2001:db8::2/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.2
set protocols ripng group ripng-group neighbor fe-1/2/1.5
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 description to-R2
set interfaces fe-1/2/0 unit 6 family inet6 address 2001:db8:0:4::/64 eui-64
set interfaces lo0 unit 3 family inet6 address 2001:db8::3/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.6
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

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 a basic RIPng network:

1. Configure the network interfaces.

   Use the `eui-64` statement to automatically generate the host portion of the interface address and the link-local address.

   For the loopback interface, you must assign a 128-bit address.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 description to-R2
   user@R1# set fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
   user@R1# set lo0 unit 1 family inet6 address 2001:db8::1/128
   ```

2. Create the RIPng group and add the interface.

   To configure RIPng in Junos OS, you must configure a group that contains the interfaces on which RIPng is enabled. You do not need to enable RIPng on the loopback interface.

   ```
   [edit protocols ripng group ripng-group]
   user@R1# set neighbor fe-1/2/0.1
   ```

3. Create the routing policy to advertise both direct and RIPng-learned routes.

```
[edit policy-options policy-statement advertise-routes-through-ripng term 1]
user@R1# set from protocol direct
user@R1# set from protocol ripng
user@R1# set then accept
```

4. Apply the routing policy.

   In Junos OS, you can only apply RIPng export policies at the group level.

```
[edit protocols ripng group ripng-group]
user@R1# set export advertise-routes-through-ripng
```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        description to-R2;
        family inet6 {
            address 2001:db8:0:1::/64 {
                eui-64;
            }
        }
    }
}
lo0 {
    unit 1 {
        family inet6 {
            address 2001:db8::1/128;
        }
```

```
    }
}
```

```
user@R1# show protocols
ripng {
    group ripng-group {
        export advertise-routes-through-ripng;
        neighbor fe-1/2/0.1;
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-ripng {
    term 1 {
        from protocol [ direct ripng ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Checking the Routing Table

Purpose

Verify that the routing table is populated with the expected routes.

Action

From operational mode, enter the `show route protocol ripng` command.

```
user@R1> show route protocol ripng
inet6.0: 12 destinations, 12 routes (12 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::2/128    *[RIPng/100] 3d 19:24:43, metric 2, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8::3/128    *[RIPng/100] 3d 19:24:40, metric 3, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8:0:2::/64  *[RIPng/100] 3d 19:24:43, metric 2, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8:0:3::/64  *[RIPng/100] 3d 19:24:43, metric 2, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8:0:4::/64  *[RIPng/100] 3d 19:24:40, metric 3, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
ff02::9/128        *[RIPng/100] 3d 19:24:47, metric 1
                         MultiRecv
```

Meaning

The output shows that the routes have been learned from Device R2 and Device R3.

If you were to delete the **from protocol ripng** condition in the routing policy on Device R2, the remote routes from Device R3 would not be learned on Device R1.

Checking the Interface Addresses

Purpose

Verify that the `eui-64` statement automatically generated the host portion of the interface address and the link-local address.

Action

From operational mode, enter the `show interfaces terse` command.

```
user@R1> show interfaces terse
Interface               Admin Link Proto    Local                 Remote
fe-1/2/0
fe-1/2/0.1              up    up   inet6    2001:db8:0:1:2a0:a514:0:14c/64
                                            fe80::2a0:a514:0:14c/64
lo0
lo0.1                   up    up   inet6    2001:db8::1
                                            fe80::2a0:a50f:fc56:14c
```

Meaning

The output shows that the interface address on fe-1/2/0.1 includes both the network portion (2001:db8:0:1) and the host portion (2a0:a514:0:14c).

Also, link-local (fe80) addresses are assigned to interfaces fe-1/2/0.1 and lo0.1.

Looking at the Routes That Device R1 Is Advertising to Device R2

Purpose

Verify that Device R1 is sending the expected routes.

Action

From operational mode, enter the `show route advertising-protocol ripng` command, using Device R1's link-local address as the neighbor address.

```
user@R1> show route advertising-protocol ripng fe80::2a0:a514:0:14c
inet6.0: 12 destinations, 12 routes (12 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::1/128    *[Direct/0] 3d 19:45:55
                    >    via lo0.1
2001:db8:0:1::/64  *[Direct/0] 3d 19:45:55
                    >    via fe-1/2/0.1
```

Meaning

Device R1 is sending routes to its directly connected networks.

Verifying the RIPng-Enabled Interfaces

Purpose

Verify that all RIPng-enabled Interfaces are available and active.

Action

From operational mode, enter the `show ripng neighbor` command.

```
user@R1> show ripng neighbor
                    Source                           Dest            In
Neighbor     State  Address                          Address  Send Recv Met
--------     -----  -------                          -------  ---- ---- ---
fe-1/2/0.1      Up fe80::2a0:a514:0:14c              ff02::9  yes  yes  1
```

Meaning

The output shows that the RIPng-enabled interface on Device R1 is operational.

The output also shows the link-local address that is assigned to Device R2's directly connected link-local interface.

In general for this command, the output shows a list of the RIPng neighbors that are configured on the device. Verify the following information:

- Each configured interface is present. Interfaces are listed in alphabetical order.
- Each configured interface is up. The state of the interface is listed in the **State** column. A state of **Up** indicates that the link is passing RIPng traffic. A state of **Dn** indicates that the link is not passing RIPng traffic. In a point-to-point link, this state generally means that either the end point is not configured for RIPng or the link is unavailable.

Looking at the Routes That Device R1 Is Receiving from Device R2

Purpose

Verify that Device R1 is receiving the expected routes.

Action

From operational mode, enter the `show route receive-protocol ripng` command, using Device R2’s directly connected link-local interface address as the neighbor address.

```
user@R1> show route receive-protocol ripng fe80::2a0:a514:0:24c
inet6.0: 12 destinations, 12 routes (12 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::2/128    *[RIPng/100] 3d 19:58:09, metric 2, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8::3/128    *[RIPng/100] 3d 19:58:06, metric 3, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8:0:2::/64  *[RIPng/100] 3d 19:58:09, metric 2, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8:0:3::/64  *[RIPng/100] 3d 19:58:09, metric 2, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8:0:4::/64  *[RIPng/100] 3d 19:58:06, metric 3, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
```

Meaning

Device R1 is receiving from Device R2 all of Device R2’s directly connected networks. Device R1 is also receiving from Device R2 all of Device R3’s directly connected networks, which Device R2 learned from Device R3 through RIPng.

Verifying the Exchange of RIPng Messages

Purpose

Verify that RIPng messages are being sent and received on all RIPng-enabled interfaces.

Action

From operational mode, enter the `show ripng statistics` command.

```
user@R1> show ripng statistics
RIPng info: port 521; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              5              0              0              0
```

```
fe-1/2/0.1:  5 routes learned; 2 routes advertised; timeout 180s; update interval 30s
Counter                          Total    Last 5 min  Last minute
-------                    -----------  -----------  -----------
Updates Sent                     11632           10            2
Triggered Updates Sent               0            0            0
Responses Sent                       0            0            0
Bad Messages                         0            0            0
Updates Received                 11634           11            2
Bad Route Entries                    0            0            0
Updates Ignored                      0            0            0
RIPng Requests Received              1            0            0
RIPng Requests Ignored               0            0            0
```

Meaning

The output shows the number of RIPng routes learned. It also shows the number of RIPng updates sent and received on the RIPng-enabled interfaces. Verify the following information:

- The number of RIPng routes learned matches the number of expected routes learned. Subnets learned by direct connectivity through an outgoing interface are not listed as RIPng routes.
- RIPng updates are being sent on each RIPng-enabled interface. If no updates are being sent, the routing policy might not be configured to export routes.
- RIPng updates are being received on each RIPng-enabled interface. If no updates are being received, the routing policy might not be configured to export routes on the host connected to that subnet. The lack of updates might also indicate an authentication error.

Verifying Reachability of All Hosts in the RIPng Network

Purpose

By using the traceroute command on each loopback address in the network, verify that all hosts in the RIPng network are reachable from each Juniper Networks device.

Action

From operational mode, enter the `traceroute` command.

```
user@R1> traceroute 2001:db8::3
traceroute6 to 2001:db8::3 (2001:db8::3) from 2001:db8:0:1:2a0:a514:0:14c, 64 hops max, 12 byte
```

```
packets
 1  2001:db8:0:2:2a0:a514:0:24c (2001:db8:0:2:2a0:a514:0:24c)  8.881 ms  1.175 ms  1.101 ms
 2  2001:db8::3 (2001:db8::3)  1.544 ms  2.445 ms  2.043 ms
```

Meaning

Each numbered row in the output indicates a routing hop in the path to the host. The three-time increments indicate the round-trip time (RTT) between the device and the hop for each traceroute packet.

To ensure that the RIPng network is healthy, verify the following information:

- The final hop in the list is the host you want to reach.
- The number of expected hops to the host matches the number of hops in the traceroute output. The appearance of more hops than expected in the output indicates that a network segment is probably unreachable. It might also indicate that the incoming or outgoing metric on one or more hosts has been set unexpectedly.

RELATED DOCUMENTATION

RIPng Overview | 8

RIPng Import Policy

IN THIS SECTION

Understanding RIPng Import Policies to Filter Routes

The default RIPng import policy is to accept all received RIPng routes that pass a validity check. To filter routes being imported by the local routing device from its neighbors, include the `import` statement and list the names of one or more policies to be evaluated. If you specify more than one policy, they are evaluated in order (first to last) and the first matching policy is applied to the route. If no match is found, the local routing device does not import any routes.

Example: Applying Policies to RIPng Routes Imported from Neighbors

IN THIS SECTION

This example shows how to configure an import policy in a RIPng network.

Requirements

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

Overview

IN THIS SECTION

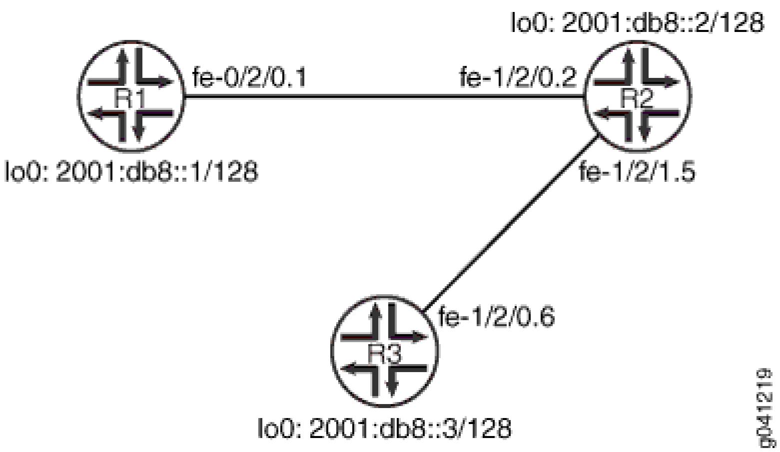
In this example, Device R2 has several extra loopback interface addresses configured to simulate additional networks.

Device R1 has an import policy that accepts the fe80::/64 and 2001:db8::/64 routes and rejects all other routes. This means that the extra networks advertised by Device R2 are not accepted into Device R1's routing table.

An export policy is also shown because an export policy is required as part of the minimum configuration for RIPng.

Figure 19 on page 147 shows the topology used in this example.

Figure 19: RIPng Import Policy Network Topology



"CLI Quick Configuration" on page 148 shows the configuration for all of the devices in Figure 19 on page 147. The section "No Link Title" on page 149 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 description to-R2
set interfaces fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces lo0 unit 1 family inet6 address 2001:db8::1/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.1 import ripng-import
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
set policy-options policy-statement ripng-import term 1 from route-filter fe80::/64 orlonger
set policy-options policy-statement ripng-import term 1 from route-filter 2001:db8::/64 orlonger
set policy-options policy-statement ripng-import term 1 then accept
set policy-options policy-statement ripng-import term 2 then reject
```

Device R2

```
set interfaces fe-1/2/0 unit 2 description to-R1
set interfaces fe-1/2/0 unit 2 family inet6 address 2001:db8:0:2::/64 eui-64
set interfaces fe-1/2/1 unit 5 description to-R3
set interfaces fe-1/2/1 unit 5 family inet6 address 2001:db8:0:3::/64 eui-64
set interfaces lo0 unit 2 family inet6 address 2001:db8::2/128
set interfaces lo0 unit 2 family inet6 address 2002:db8::2/128
set interfaces lo0 unit 2 family inet6 address 2002:db9::2/128
set interfaces lo0 unit 2 family inet6 address 2002:db7::2/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.2
set protocols ripng group ripng-group neighbor fe-1/2/1.5
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Device R3

```
set  interfaces fe-1/2/0 unit 6 description to-R2
set  interfaces fe-1/2/0 unit 6 family inet6 address 2001:db8:0:4::/64 eui-64
set  interfaces lo0 unit 3 family inet6 address 2001:db8::3/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.6
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

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 a RIPng import policy:

1. Configure the network interfaces.

   This example shows multiple loopback interface addresses to simulate attached networks.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 description to-R2
   user@R1# set fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
   user@R1# set lo0 unit 1 family inet6 address 2001:db8::1/128
   ```

2. Create the RIPng group and add the interface.

   To configure RIPng in Junos OS, you must configure a group that contains the interfaces on which RIPng is enabled. You do not need to enable RIPng on the loopback interface.

   ```
   [edit protocols ripng group ripng-group]
   user@R1# set neighbor fe-1/2/0.1
   ```

3. Create the routing policy to advertise both direct and RIPng-learned routes.

   ```
   [edit policy-options policy-statement advertise-routes-through-ripng term 1]
   user@R1# set from protocol direct
   ```

```
user@R1# set from protocol ripng
user@R1# set then accept
```

4. Apply the routing policy.

   In Junos OS, you can only apply RIPng export policies at the group level.

```
[edit protocols ripng group ripng-group]
user@R1# set export advertise-routes-through-ripng
```

5. Configure the import policy.

```
[edit policy-options policy-statement ripng-import]
user@R1# set term 1 from route-filter fe80::/64 orlonger
user@R1# set term 1 from route-filter 2001:db8::/64 orlonger
user@R1# set term 1 then accept
user@R1# set term 2 then reject
```

6. Apply the import policy.

```
[edit protocols ripng group ripng-group]
user@R1# set neighbor fe-1/2/0.1 import ripng-import
```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        description to-R2;
        family inet6 {
            address 2001:db8:0:1::/64 {
                eui-64;
            }
        }
    }
```

```
}
lo0 {
    unit 1 {
        family inet6 {
            address 2001:db8::1/128;
        }
    }
}
```

```
user@R1# show protocols
ripng {
    group ripng-group {
        export advertise-routes-through-ripng;
        neighbor fe-1/2/0.1 {
            import ripng-import;
        }
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-ripng {
    term 1 {
        from protocol [ direct ripng ];
        then accept;
    }
}
policy-statement ripng-import {
    term 1 {
        from {
            route-filter fe80::/64 orlonger;
            route-filter 2001:db8::/64 orlonger;
        }
        then accept;
    }
    term 2 {
        then reject;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Looking at the Neighbor Addresses for Device R2

Purpose

Determine the neighbor address that Device R2 is using for Device R1.

Action

From operational mode, enter the `show ripng neighbor` command.

```
user@R2> show ripng neighbor fe-1/2/0.2

                    Source                          Dest              In
Neighbor     State  Address                         Address  Send Recv Met
--------     -----  -------                         -------  ---- ---- ---
fe-1/2/0.2      Up fe80::2a0:a514:0:24c             ff02::9  yes   yes  1
```

Meaning

Device R2 is using the fe80::2a0:a514:0:24c address to send routes to Device R1.

Looking at the Routes That Device R2 Is Advertising to Device R1

Purpose

Verify that Device R2 is sending the expected routes.

Action

From operational mode, enter the `show route advertising-protocol ripng` command.

```
user@R2> show route advertising-protocol ripng fe80::2a0:a514:0:24c
inet6.0: 17 destinations, 18 routes (17 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::2/128    *[Direct/0] 3d 22:00:34
                    >    via lo0.2
2001:db8::3/128    *[RIPng/100] 3d 21:47:00, metric 2, tag 0
                    > to fe80::2a0:a514:0:64c via fe-1/2/1.5
2001:db8:0:2::/64  *[Direct/0] 3d 22:00:34
                    >    via fe-1/2/0.2
2001:db8:0:3::/64  *[Direct/0] 3d 22:00:34
                    >    via fe-1/2/1.5
2001:db8:0:4::/64  *[RIPng/100] 3d 21:47:00, metric 2, tag 0
                    > to fe80::2a0:a514:0:64c via fe-1/2/1.5
2002:db7::2/128    *[Direct/0] 00:29:05
                    >    via lo0.2
2002:db8::2/128    *[Direct/0] 00:31:49
                    >    via lo0.2
2002:db9::2/128    *[Direct/0] 00:29:05
                    >    via lo0.2
```

Meaning

Device R2 is sending the extra loopback interface /128 routes to Device R1.

Looking at the Routes That Device R1 Is Receiving from Device R2

Purpose

Verify that Device R1 is receiving the expected routes.

Action

From operational mode, enter the `show route receive-protocol ripng` command.

```
user@R1> show route receive-protocol ripng fe80::2a0:a514:0:24c

inet6.0: 9 destinations, 9 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::2/128    *[RIPng/100] 3d 21:55:49, metric 2, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8::3/128    *[RIPng/100] 3d 21:55:46, metric 3, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
```

Meaning

The output shows that the extra loopback interface addresses are excluded.

Checking the Routing Table

Purpose

Verify that the routing table is populated with the expected routes.

Action

From operational mode, enter the `show route protocol ripng` command.

```
user@R1> show route protocol ripng
inet6.0: 9 destinations, 9 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::2/128    *[RIPng/100] 3d 22:01:40, metric 2, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
2001:db8::3/128    *[RIPng/100] 3d 22:01:37, metric 3, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
ff02::9/128        *[RIPng/100] 00:00:08, metric 1
                         MultiRecv
```

Meaning

The output shows that the routes have been learned from Device R2 and Device R3.

If you delete or deactivate the import policy, the routing table contains the extra loopback interface routes.

Example: Testing a Routing Policy with Complex Regular Expressions

IN THIS SECTION

This example shows how to test a routing policy using the `test policy` command to ensure that the policy produces the results that you expect before you apply it in a production environment. Regular expressions, especially complex ones, can be tricky to get right. This example shows how to use the `test policy` command to make sure that your regular expressions have the intended effect.

Requirements

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

Overview

IN THIS SECTION

This example shows two routing devices with an external BGP (EBGP) connection between them. Device R2 uses the BGP session to send customer routes to Device R1. These static routes have multiple community values attached.

```
user@R2> show route match-prefix 172.16.* detail

inet.0: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
172.16.1.0/24 (1 entry, 1 announced)
        *Static Preference: 5
                Next hop type: Reject
                Address: 0x8fd0dc4
                Next-hop reference count: 8
                State: <Active Int Ext>
                Local AS: 64511
                Age: 21:32:13
                Validation State: unverified
                Task: RT
                Announcement bits (1): 0-KRT
                AS path: I
                Communities: 64510:1 64510:10 64510:11 64510:100 64510:111

172.16.2.0/24 (1 entry, 1 announced)
        *Static Preference: 5
                Next hop type: Reject
                Address: 0x8fd0dc4
                Next-hop reference count: 8
                State: <Active Int Ext>
                Local AS: 64511
                Age: 21:32:13
                Validation State: unverified
                Task: RT
                Announcement bits (1): 0-KRT
                AS path: I
                Communities: 64510:2 64510:20 64510:22 64510:200 64510:222

172.16.3.0/24 (1 entry, 1 announced)
        *Static Preference: 5
                Next hop type: Reject
                Address: 0x8fd0dc4
                Next-hop reference count: 8
                State: <Active Int Ext>
                Local AS: 64511
```

```
                Age: 21:32:13
                Validation State: unverified
                Task: RT
                Announcement bits (1): 0-KRT
                AS path: I
                Communities: 64510:3 64510:30 64510:33 64510:300 64510:333

172.16.4.0/24 (1 entry, 1 announced)
        *Static Preference: 5
                Next hop type: Reject
                Address: 0x8fd0dc4
                Next-hop reference count: 8
                State: <Active Int Ext>
                Local AS: 64511
                Age: 21:32:13
                Validation State: unverified
                Task: RT
                Announcement bits (1): 0-KRT
                AS path: I
                Communities: 64510:4 64510:40 64510:44 64510:400 64510:444
```

To test a complex regular expression, Device R2 has a policy called `test-regex` that locates routes. The policy is configured like this:

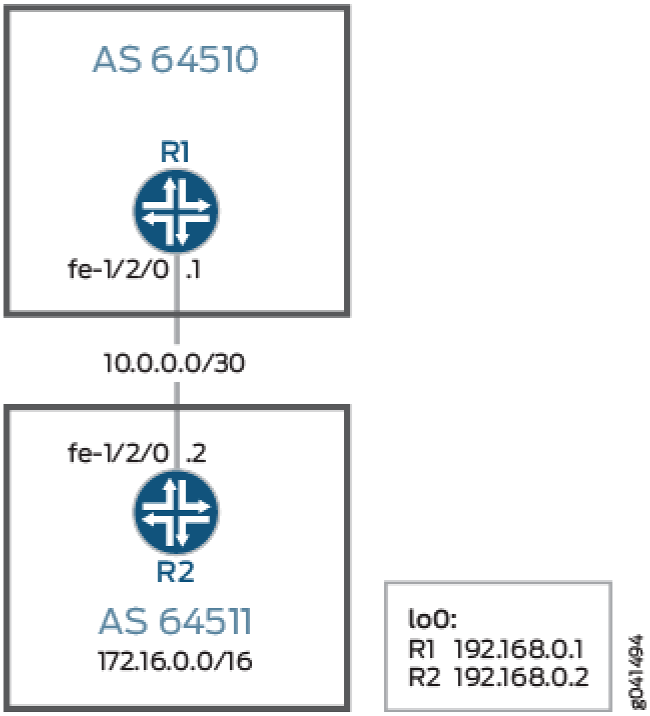
```
policy-statement test-regex {
    term find-routes {
        from community complex-regex;
        then accept;
    }
    term reject-the-rest {
        then reject;
    }
}
community complex-regex members "^64510:[13].*$";
```

This regular expression matches community values beginning with either 1 or 3.

Topology

Figure 20 on page 158 shows the sample network.

Figure 20: Routing Policy Test for Complex Regular Expressions



"CLI Quick Configuration" on page 158 shows the configuration for all of the devices in Figure 20 on page 158.

The section "No Link Title" on page 160 describes the steps on Device R2.

Configuration

IN THIS SECTION

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.1/30
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set protocols bgp group ext type external
set protocols bgp group ext peer-as 64511
set protocols bgp group ext neighbor 10.0.0.2
set routing-options router-id 192.168.0.1
set routing-options autonomous-system 64510
```

Device R2

```
set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.2/30
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set protocols bgp group ext type external
set protocols bgp group ext peer-as 64510
set protocols bgp group ext neighbor 10.0.0.1
set policy-options policy-statement send-static term 1 from protocol static
set policy-options policy-statement send-static term 1 then accept
set policy-options policy-statement send-static term 2 then reject
set policy-options policy-statement test-regex term find-routes from community complex-regex
set policy-options policy-statement test-regex term find-routes then accept
set policy-options policy-statement test-regex term reject-the-rest then reject
set policy-options community complex-regex members "^64510:[13].*$"
set routing-options static route 172.16.1.0/24 reject
set routing-options static route 172.16.1.0/24 community 64510:1
set routing-options static route 172.16.1.0/24 community 64510:10
set routing-options static route 172.16.1.0/24 community 64510:11
set routing-options static route 172.16.1.0/24 community 64510:100
set routing-options static route 172.16.1.0/24 community 64510:111
set routing-options static route 172.16.2.0/24 reject
set routing-options static route 172.16.2.0/24 community 64510:2
set routing-options static route 172.16.2.0/24 community 64510:20
set routing-options static route 172.16.2.0/24 community 64510:22
set routing-options static route 172.16.2.0/24 community 64510:200
set routing-options static route 172.16.2.0/24 community 64510:222
set routing-options static route 172.16.3.0/24 reject
set routing-options static route 172.16.3.0/24 community 64510:3
set routing-options static route 172.16.3.0/24 community 64510:30
set routing-options static route 172.16.3.0/24 community 64510:33
set routing-options static route 172.16.3.0/24 community 64510:300
```

```
set routing-options static route 172.16.3.0/24 community 64510:333
set routing-options static route 172.16.4.0/24 reject
set routing-options static route 172.16.4.0/24 community 64510:4
set routing-options static route 172.16.4.0/24 community 64510:40
set routing-options static route 172.16.4.0/24 community 64510:44
set routing-options static route 172.16.4.0/24 community 64510:400
set routing-options static route 172.16.4.0/24 community 64510:444
set routing-options router-id 192.168.0.2
set routing-options autonomous-system 64511
```

Procedure

Step-by-Step Procedure

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

To configure Device R2:

1. Configure the interfaces.

```
[edit interfaces]
user@R2# set fe-1/2/0 unit 0 family inet address 10.0.0.2/30
user@R2# set lo0 unit 0 family inet address 192.168.0.2/32
```

2. Configure BGP.

   Apply the import policy to the BGP peering session with Device R2.

```
[edit protocols bgp group ext]
user@R2# set type external
user@R2# set peer-as 64510
user@R2# set neighbor 10.0.0.1
```

3. Configure the routing policy that sends static routes.

```
[edit policy-options policy-statement send-static]
user@R2# set term 1 from protocol static
```

```
user@R2# set term 1 then accept
user@R2# set term 2 then reject
```

4. Configure the routing policy that tests a regular expression.

```
[edit policy-options policy-statement test-regex]
user@R2# set term find-routes from community complex-regex
user@R2# set term find-routes then accept
user@R2# set term reject-the-rest then reject
[edit policy-options community]
user@R2# set complex-regex members "^64510:[13].*$"
```

5. Configure the static routes and attaches community values.

```
[edit routing-options static route 172.16.1.0/24]
user@R2# set reject
user@R2# set community [ 64510:1 64510:10 64510:11 64510:100 64510:111 ]
[edit routing-options static route 172.16.2.0/24]
user@R2# set reject
user@R2# set community [ 64510:2 64510:20 64510:22 64510:200 64510:222 ]
[edit routing-options static route 172.16.3.0/24]
user@R2# set reject
user@R2# set community [ 64510:3 64510:30 64510:33 64510:300 64510:333 ]
[edit routing-options static route 172.16.4.0/24]
user@R2# set reject
user@R2# set community [ 64510:4 64510:40 64510:44 64510:400 64510:444 ]
```

6. Configure the autonomous system (AS) number and the router ID.

   This affects Device R2's routing table, and as no impact on Device R1 and Device R3.

```
[edit routing-options ]
user@R2# set router-id 192.168.0.2
user@R2# set autonomous-system 64511
```

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@R2# show interfaces
fe-1/2/0 {
    unit 0 {
        family inet {
            address 10.0.0.2/30;
        }
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.0.2/32;
        }
    }
}
```

```
user@R2# show protocols
bgp {
    group ext {
        type external;
        peer-as 64510;
        neighbor 10.0.0.1;
    }
}
```

```
user@R2# show policy-options
policy-statement send-static {
    term 1 {
        from protocol static;
        then accept;
    }
    term 2 {
        then reject;
```

```
    }
}
policy-statement test-regex {
    term find-routes {
        from community complex-regex;
        then accept;
    }
    term reject-the-rest {
        then reject;
    }
}
community complex-regex members "^64510:[13].*$";
```

```
user@R2# show routing-options
static {
    route 172.16.1.0/24 {
        reject;
        community [ 64510:1 64510:10 64510:11 64510:100 64510:111 ];
    }
    route 172.16.2.0/24 {
        reject;
        community [ 64510:2 64510:20 64510:22 64510:200 64510:222 ];
    }
    route 172.16.3.0/24 {
        reject;
        community [ 64510:3 64510:30 64510:33 64510:300 64510:333 ];
    }
    route 172.16.4.0/24 {
        reject;
        community [ 64510:4 64510:40 64510:44 64510:400 64510:444 ];
    }
}
router-id 192.168.0.2;
autonomous-system 64511;
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Test to See Which Communities Match the Regular Expression

Purpose

You can test the regular expression and its policy by using the `test policy`*`policy-name`* command.

Action

1. On Device R2, run the `test policy test-regex 0/0` command.

```
user@R2> test policy test-regex 0/0


inet.0: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.1.0/24      *[Static/5] 1d 00:32:50
                      Reject
172.16.3.0/24      *[Static/5] 1d 00:32:50
                      Reject

Policy test-regex: 2 prefix accepted, 5 prefix rejected
```

2. On Device R2, change the regular expression to match a community value containing any number of instances of the digit 2.

```
[edit policy-options community complex-regex]
user@R2# delete members "^64510:[13].*$"
```

```
user@R2# set members "^65020:2+$"
user@R2# commit
```

3. On Device R2, rerun the `test policy test-regex 0/0` command.

```
user@R2> test policy test-regex 0/0

inet.0: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.2.0/24      *[Static/5] 1d 00:31:36
                      Reject

Policy test-regex: 1 prefix accepted, 6 prefix rejected
```

Meaning

The 172.16.1.0 /24 and 172.16.3.0/24 routes both have communities attached that match the ^64510:[13].*$ expression. The 172.16.2.0/24 route has communities that match the ^65020:2+$ expression.

RELATED DOCUMENTATION

Understanding Routing Policy Tests

Understanding How to Define BGP Communities and Extended Communities

Understanding AS Path Regular Expressions for Use as Routing Policy Match Conditions

Traffic Control in a RIPng Network

IN THIS SECTION

Understanding RIPng Traffic Control with Metrics for Optimizing the Path Cost

To tune a RIPng network and to control traffic flowing through the network, you increase or decrease the cost of the paths through the network. RIPng provides two ways to modify the path cost: an incoming metric and an outgoing metric, which are each set to 1 by default. In other words, by default, the metric of routes that RIPng imports from a neighbor or exports to a neighbor is incremented by 1. These routes include those learned from RIPng as well as those learned from other protocols. The metrics are attributes that specify the cost of any route advertised through a host. By increasing or decreasing the metrics—and thus the cost—of links throughout the network, you can control packet transmission across the network.

The incoming metric modifies the cost of an individual segment when a route across the segment is imported into the routing table. For example, if you set the incoming metric on the segment to **3**, the individual segment cost along the link is changed from **1** to **3**. The increased cost affects all route calculations through that link. Other routes that were previously excluded because of a high hop count might now be selected into the router's forwarding table.

The outgoing metric modifies the path cost for all the routes advertised out of a particular interface. Unlike the incoming metric, the outgoing metric modifies the routes that other routers are learning and thereby controls the way they send traffic.

If an exported route was learned from a member of the same RIPng group, the metric associated with that route is the normal RIPng metric. For example, a RIPng route with a metric of 5 learned from a neighbor configured with an incoming metric of 2 is advertised with a combined metric of 7 when advertised to neighbors in the same group. However, if this route was learned from a RIPng neighbor in a different group or from a different protocol, the route is advertised with the metric value configured in the outgoing metric for that group.

You might want to increase the metric of routes to decrease the likelihood that a particular route is selected and installed in the routing table. This process is sometimes referred to as *route poisoning*. Some reasons that you might want to poison a route are that the route is relatively expensive to use, or it has relatively low bandwidth.

A route with a higher metric than another route becomes the active route only when the lower-metric route becomes unavailable. In this way, the higher-metric route serves as a backup path.

One way to increase the metric of imported routes is to configure an import policy. Another way is to include the `metric-in` statement in the RIPng neighbor configuration. One way to increase the metric of export routes is to configure an export policy. Another way is to include the `metric-out` statement in the RIPng neighbor configuration.

Example: Configuring the Metric Value Added to Imported RIPng Routes to Control the Route Selection Process

IN THIS SECTION

This example shows how to change the default metric to be added to incoming routes to control the route selection process.

Requirements

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

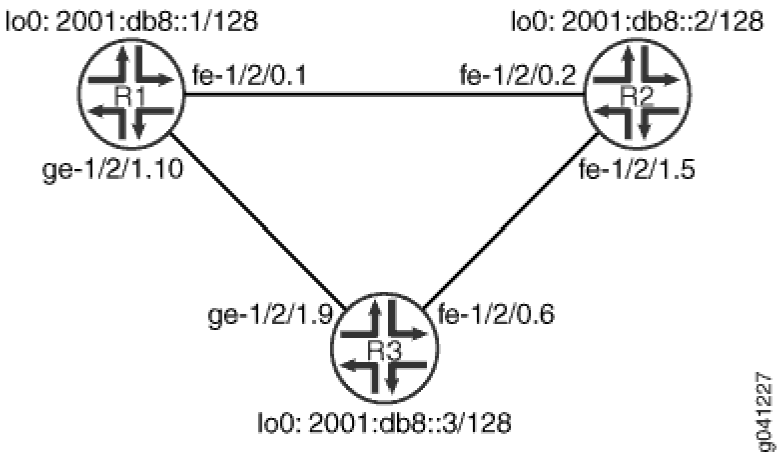
Overview

IN THIS SECTION

Normally, when multiple routes are available, RIPng selects the route with the lowest hop count. Changing the default metric enables you to control the route selection process such that a route with a higher hop count can be preferred over of a route with a lower hop count.

Figure 21 on page 168 shows the topology used in this example.

Figure 21: RIPng Incoming Metrics Network Topology



Device R1 has two potential paths to reach 2001:db8::2/128. The default behavior is to send traffic out the 2001:db8:0:1::/64 interface facing Device R2. Suppose, though, that the path through Device R3 is less expensive to use or has higher bandwidth links. This example shows how to use the `metric-in` statement to ensure that Device R1 uses the path through Device R3 to reach 2001:db8::2/128. "CLI Quick Configuration" on page 168 shows the configuration for all of the devices in Figure 21 on page 168. The section "No Link Title" on page 170 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 description to-R2
set interfaces fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces ge-1/2/1 unit 10 description to-R3
set interfaces ge-1/2/1 unit 10 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces lo0 unit 1 family inet6 address 2001:db8::1/128
set protocols ripng group primary export advertise-routes-through-ripng
set protocols ripng group primary neighbor ge-1/2/1.10
set protocols ripng group secondary export advertise-routes-through-ripng
set protocols ripng group secondary neighbor fe-1/2/0.1 metric-in 4
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet6 address 2001:db8:0:2::/64 eui-64
set interfaces fe-1/2/1 unit 5 description to-R3
set interfaces fe-1/2/1 unit 5 family inet6 address 2001:db8:0:3::/64 eui-64
set interfaces lo0 unit 2 family inet6 address 2001:db8::2/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.2
set protocols ripng group ripng-group neighbor fe-1/2/1.5
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 family inet6 address 2001:db8:0:4::/64 eui-64
set interfaces ge-1/2/1 unit 9 description to-R1
set interfaces ge-1/2/1 unit 9 family inet address 10.0.0.9/30
set interfaces ge-1/2/1 unit 9 family inet6 address 2001:db8:0:6::/64 eui-64
set interfaces lo0 unit 3 family inet6 address 2001:db8::3/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.6
set protocols ripng group ripng-group neighbor ge-1/2/1.9
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
```

```
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

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 a RIPng metrics:

1. Configure the network interfaces.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 description to-R2
   user@R1# set fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
   user@R1# set ge-1/2/1 unit 10 description to-R3
   user@R1# set ge-1/2/1 unit 10 family inet6 address 2001:db8:0:5::/64 eui-64
   user@R1# set lo0 unit 1 family inet6 address 2001:db8::1/128
   ```

2. Create the RIPng groups and add the interfaces.

   To configure RIPng in Junos OS, you must configure one or more groups that contain the interfaces on which RIPng is enabled. You do not need to enable RIPng on the loopback interface.

   For the interface that is facing Device R2, the **metric-in 4** setting causes this route to be less likely to be chosen as the active route.

   ```
   [edit protocols ripng]
   user@R1# set group primary neighbor ge-1/2/1.10
   user@R1# set group secondary neighbor fe-1/2/0.1 metric-in 4
   ```

3. Create the routing policy to advertise both direct and RIPng-learned routes.

   ```
   [edit policy-options policy-statement advertise-routes-through-ripng term 1]
   user@R1# set from protocol direct
   user@R1# set from protocol ripng
   user@R1# set then accept
   ```

4. Apply the routing policy.

In Junos OS, you can only apply RIPng export policies at the group level.

```
[edit protocols ripng]
user@R1# set group primary export advertise-routes-through-ripng
user@R1# set group secondary export advertise-routes-through-ripng
```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        description to-R2;
        family inet6 {
            address 2001:db8:0:1::/64 {
                eui-64;
            }
        }
    }
}
ge-1/2/1 {
    unit 10 {
        description to-R3;
        family inet6 {
            address 2001:db8:0:5::/64 {
                eui-64;
            }
        }
    }
}
lo0 {
    unit 1 {
        family inet6 {
            address 2001:db8::1/128;
        }
```

```
    }
}
```

```
user@R1# show protocols
ripng {
    group primary {
        export advertise-routes-through-ripng;
        neighbor ge-1/2/0.10;
    }
    group secondary {
        export advertise-routes-through-ripng;
        neighbor fe-1/2/0.1 {
            metric-in 4;
        }
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-ripng {
    term 1 {
        from protocol [ direct ripng ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying That the Expected Route Is Active

Purpose

Make sure that Device R1 uses the path through Device R3 to reach 2001:db8:0:2:2a0:a514:0:24c/128.

Action

From operational mode, enter the `show route 2001:db8:0:2:2a0:a514:0:24c` command.

```
user@R1> show route 2001:db8:0:2:2a0:a514:0:24c
inet6.0: 16 destinations, 17 routes (16 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8:0:2::/64  *[RIPng/100] 01:54:35, metric 3, tag 0
                    > to fe80::2a0:a514:0:94c via ge-1/2/1.10
```

Meaning

The **to fe80::2a0:a514:0:94c via ge-1/2/1.10** output shows that Device R1 uses the path through Device R3 to reach 2001:db8:0:2:2a0:a514:0:24c/128. The metric for this route is 3.

Removing the metric-in Statement

Purpose

Delete or deactivate the metric-in statement to see what happens to the 2001:db8:0:2:2a0:a514:0:24c/128 route.

Action

1. From configuration mode, deactivate the `metric-in` statement.

   ```
   [edit protocols ripng group secondary neighbor fe-1/2/0.1]
   user@R1# deactivate metric-in
   user@R1# commit
   ```

2. From operational mode, enter the `show route 2001:db8:0:2:2a0:a514:0:24c` command.

```
user@R1> show route 2001:db8:0:2:2a0:a514:0:24c
inet6.0: 16 destinations, 17 routes (16 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8:0:2::/64  *[RIPng/100] 00:00:02, metric 2, tag 0
                    > to fe80::2a0:a514:0:24c via fe-1/2/0.1
```

Meaning

The **to fe80::2a0:a514:0:24c via fe-1/2/0.1** output shows that Device R1 uses the path through Device R2 to reach 2001:db8:0:2:2a0:a514:0:24c/128. The metric for this route is 2.

RIPng Timers

IN THIS SECTION

Example: Configuring RIPng Update Interval

IN THIS SECTION

This example shows how to configure the RIPng update interval and how to monitor the impact of the change.

Requirements

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

Overview

IN THIS SECTION

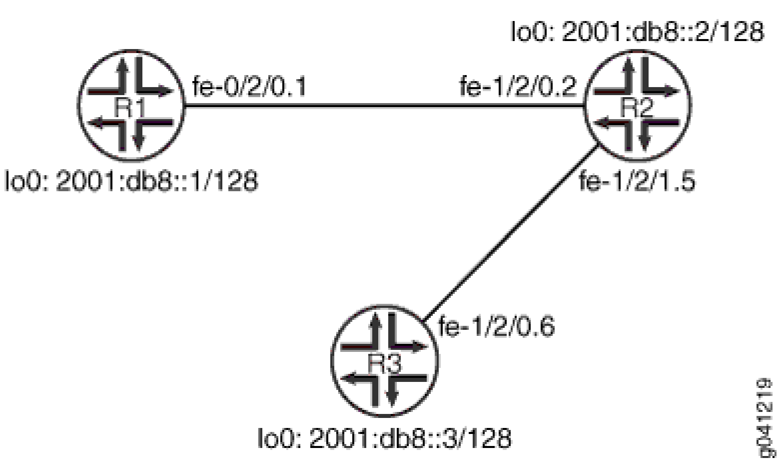
In this example, Device R2 has an update interval of 60 seconds for its neighbor Device R1, and an update interval of 10 seconds for its neighbor Device R3.

This example is not necessarily practical, but it is shown for demonstration purposes. Generally, we recommend against changing the RIPng timers, unless the effects of a change are well understood. Normally, the default values are best left in effect for standard operations.

An export policy is also shown because an export policy is required as part of the minimum configuration for RIPng.

Figure 22 on page 175 shows the topology used in this example.

Figure 22: RIPng Timers Network Topology



"CLI Quick Configuration" on page 176 shows the configuration for all of the devices in Figure 22 on page 175. The section "No Link Title" on page 177 describes the steps on Device R2.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 description to-R2
set interfaces fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces lo0 unit 1 family inet6 address 2001:db8::1/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.1
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 description to-R1
set interfaces fe-1/2/0 unit 2 family inet6 address 2001:db8:0:2::/64 eui-64
set interfaces fe-1/2/1 unit 5 description to-R3
set interfaces fe-1/2/1 unit 5 family inet6 address 2001:db8:0:3::/64 eui-64
set interfaces lo0 unit 2 family inet6 address 2001:db8::2/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.2 update-interval 60
set protocols ripng group ripng-group neighbor fe-1/2/1.5 update-interval 10
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
```

```
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 description to-R2
set interfaces fe-1/2/0 unit 6 family inet6 address 2001:db8:0:4::/64 eui-64
set interfaces lo0 unit 3 family inet6 address 2001:db8::3/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.6
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Step-by-Step Procedure

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

To configure the RIPng update interval:

1. Configure the network interfaces.

   This example shows multiple loopback interface addresses to simulate attached networks.

   ```
   [edit interfaces]
   user@R2# set fe-1/2/0 unit 2 description to-R1
   user@R2# set fe-1/2/0 unit 2 family inet6 address 2001:db8:0:2::/64 eui-64
   user@R2# set fe-1/2/1 unit 5 description to-R3
   user@R2# set fe-1/2/1 unit 5 family inet6 address 2001:db8:0:3::/64 eui-64
   user@R2# set lo0 unit 2 family inet6 address 2001:db8::2/128
   ```

2. Configure different update intervals for the two RIPng neighbors.

To configure RIPng in Junos OS, you must configure a group that contains the interfaces on which RIPng is enabled. You do not need to enable RIPng on the loopback interface.

```
[edit protocols ripng group ripng-group]
user@R2# set neighbor fe-1/2/0.2 update-interval 60
user@R2# set neighbor fe-1/2/1.5 update-interval 10
```

3. Create the routing policy to advertise both direct and RIPng-learned routes.

```
[edit policy-options policy-statement advertise-routes-through-ripng term 1]
user@R2# set from protocol direct
user@R2# set from protocol ripng
user@R2# set then accept
```

4. Apply the routing policy.

   In Junos OS, you can only apply RIPng export policies at the group level.

```
[edit protocols ripng group ripng-group]
user@R2# set export advertise-routes-through-ripng
```

Results

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

```
user@R2# show interfaces
fe-1/2/0 {
    unit 2 {
        description to-R1;
        family inet6 {
            address 2001:db8:0:2::/64 {
                eui-64;
            }
        }
    }
}
fe-1/2/1 {
```

```
    unit 5 {
        description to-R3;
        family inet6 {
            address 2001:db8:0:3::/64 {
                eui-64;
            }
        }
    }
}
lo0 {
    unit 2 {
        family inet6 {
            address 2001:db8::2/128;
        }
    }
}
```

```
user@R2# show protocols
ripng {
    group ripng-group {
        export advertise-routes-through-ripng;
        neighbor fe-1/2/0.2 {
            update-interval 60;
        }
        neighbor fe-1/2/1.5 {
            update-interval 10;
        }
    }
}
```

```
user@R2# show policy-options
policy-statement advertise-routes-through-ripng {
    term 1 {
        from protocol [ direct ripng ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Checking the RIPng Updates Sent by Device R2

Purpose

Make sure that the RIPng update packets are sent at the expected interval.

Action

From operational mode, enter the `show ripng statistics` command.

```
user@R2> show ripng statistics
RIPng info: port 521; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              4              0              0              0

fe-1/2/0.2:  2 routes learned; 5 routes advertised; timeout 180s; update interval 60s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                        1            1            1
Triggered Updates Sent              0            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
Updates Received                    1            0            0
Bad Route Entries                   0            0            0
Updates Ignored                     0            0            0
RIPng Requests Received             0            0            0
RIPng Requests Ignored              0            0            0
```

```
fe-1/2/1.5:  2 routes learned; 5 routes advertised; timeout 180s; update interval 10s
Counter                          Total   Last 5 min  Last minute
-------                    -----------  -----------  -----------
Updates Sent                         6            2            2
Triggered Updates Sent               0            0            0
Responses Sent                       0            0            0
Bad Messages                         0            0            0
Updates Received                     2            0            0
Bad Route Entries                    0            0            0
Updates Ignored                      0            0            0
RIPng Requests Received              0            0            0
RIPng Requests Ignored               0            0            0
```

Meaning

The **update interval** field shows that the interval is 60 seconds for its neighbor Device R1 and 10 seconds for its neighbor Device R3. The **Updates Sent** field shows that Device R2 is sending updates to Device R1 at roughly 1/6 of the rate that it is sending updates to Device R3.

Checking the RIPng Updates Received by Device R2

Purpose

Make sure that the RIPng update packets are sent at the expected interval.

Action

From operational mode, enter the `show ripng statistics` command.

```
user@R1> show ripng statistics
RIPng info: port 521; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              5              8              0              0

fe-1/2/0.1:  5 routes learned; 2 routes advertised; timeout 180s; update interval 30s
Counter                          Total   Last 5 min  Last minute
-------                    -----------  -----------  -----------
Updates Sent                         6            5            2
Triggered Updates Sent               0            0            0
Responses Sent                       0            0            0
Bad Messages                         0            0            0
```

```
Updates Received                  3           3           1
Bad Route Entries                 0           0           0
Updates Ignored                   0           0           0
RIPng Requests Received           0           0           0
RIPng Requests Ignored            0           0           0
```

Meaning

The **Updates Received** field shows the number of updates received from Device R2.

Checking the RIPng Updates Received by Device R3

Purpose

Make sure that the RIPng update packets are sent at the expected interval.

Action

From operational mode, enter the `show ripng statistics` command.

```
user@R3> show ripng statistics
RIPng info: port 521; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
              5              0              0              0


fe-1/2/0.6:  5 routes learned; 2 routes advertised; timeout 180s; update interval 30s
Counter                         Total   Last 5 min  Last minute
-------                    -----------  -----------  -----------
Updates Sent                      5           5           2
Triggered Updates Sent            0           0           0
Responses Sent                    0           0           0
Bad Messages                      0           0           0
Updates Received                 16          15           6
Bad Route Entries                 0           0           0
Updates Ignored                   0           0           0
RIPng Requests Received           0           0           0
RIPng Requests Ignored            0           0           0
```

Meaning

The **Updates Received** field shows the number of updates received from Device R2.

SEE ALSO

Understanding RIP Timers | 43

Tracing RIPng Traffic

IN THIS SECTION

Understanding RIPng Protocol Traffic Trace Operations

You can trace various RIPng protocol traffic to help debug RIP protocol issues.

To trace RIP protocol traffic, include the `traceoptions` statement at the `[edit protocols ripng]` hierarchy level:

```
traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
}
```

You can specify the following RIPng protocol-specific trace options using the `flag` statement:

- **error**—RIPng error packets
- **expiration**—RIPng route expiration processing
- **holddown**—RIPng hold-down processing

- **nsr-synchronization**—Nonstop routing synchronization events
- **packets**—All RIPng packets
- **request**—RIPng information packets
- **trigger**—RIPng triggered updates
- **update**—RIPng update packets

You can optionally specify one or more of the following flag modifiers:

- **detail**—Detailed trace information
- **receive**—Packets being received
- **send**—Packets being transmitted

> **NOTE**: Use the **detail** flag modifier with caution as this might cause the CPU to become very busy.

Global tracing options are inherited from the configuration set by the `traceoptions` statement at the `[edit routing-options]` hierarchy level. You can override the following global trace options for the RIPng protocol using the `traceoptions flag` statement included at the `[edit protocols ripng]` hierarchy level:

- **all**—All tracing operations
- **general**—All normal operations and routing table changes (a combination of the normal and route trace operations)
- **normal**—Normal events
- **policy**—Policy processing
- **route**—Routing information
- **state**—State transitions
- **task**—Routing protocol task processing
- **timer**—Routing protocol timer processing

> **NOTE**: Use the trace flag **all** with caution as this might cause the CPU to become very busy.

SEE ALSO

Example: Tracing Global Routing Protocol Operations

Example: Tracing RIPng Protocol Traffic

IN THIS SECTION

This example shows how to trace RIPng protocol operations.

Requirements

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

Overview

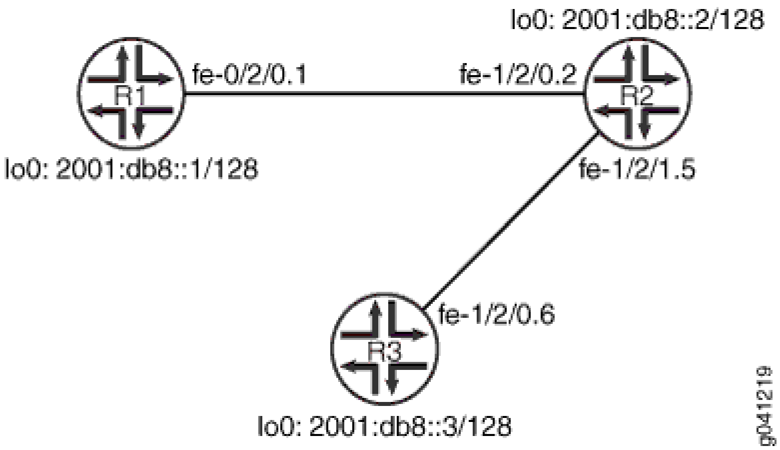
IN THIS SECTION

In this example, Device R1 is set to trace routing information updates.

An export policy is also shown because an export policy is required as part of the minimum configuration for RIPng.

Figure 23 on page 186 shows the topology used in this example.

Figure 23: RIPng Trace Operations Network Topology



"CLI Quick Configuration" on page 186 shows the configuration for all of the devices in Figure 23 on page 186. The section "No Link Title" on page 188 describes the steps on Device R1.

Topology

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device R1

```
set interfaces fe-1/2/0 unit 1 description to-R2
set interfaces fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces lo0 unit 1 family inet6 address 2001:db8::1/128
```

```
set protocols ripng traceoptions file ripng-trace-file
set protocols ripng traceoptions flag route
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.1
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Device R2

```
set interfaces fe-1/2/0 unit 2 description to-R1
set interfaces fe-1/2/0 unit 2 family inet6 address 2001:db8:0:2::/64 eui-64
set interfaces fe-1/2/1 unit 5 description to-R3
set interfaces fe-1/2/1 unit 5 family inet6 address 2001:db8:0:3::/64 eui-64
set interfaces lo0 unit 2 family inet6 address 2001:db8::2/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.2
set protocols ripng group ripng-group neighbor fe-1/2/1.5
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Device R3

```
set interfaces fe-1/2/0 unit 6 description to-R2
set interfaces fe-1/2/0 unit 6 family inet6 address 2001:db8:0:4::/64 eui-64
set interfaces lo0 unit 3 family inet6 address 2001:db8::3/128
set protocols ripng group ripng-group export advertise-routes-through-ripng
set protocols ripng group ripng-group neighbor fe-1/2/0.6
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol direct
set policy-options policy-statement advertise-routes-through-ripng term 1 from protocol ripng
set policy-options policy-statement advertise-routes-through-ripng term 1 then accept
```

Step-by-Step Procedure

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

To configure the RIPng update interval:

1. Configure the network interfaces.

   This example shows multiple loopback interface addresses to simulate attached networks.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 description to-R2
   user@R1# set fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
   user@R1# set lo0 unit 1 family inet6 address 2001:db8::1/128
   ```

2. Configure the RIPng group, and add the interface to the group.

   To configure RIPng in Junos OS, you must configure a group that contains the interfaces on which RIPng is enabled. You do not need to enable RIPng on the loopback interface.

   ```
   [edit protocols ripng group ripng-group]
   user@R1# set neighbor fe-1/2/0.1
   ```

3. Configure RIPng tracing operations.

   ```
   [edit protocols ripng traceoptions]
   user@R1# set file ripng-trace-file
   user@R1# set flag route
   ```

4. Create the routing policy to advertise both direct and RIPng-learned routes.

   ```
   [edit policy-options policy-statement advertise-routes-through-ripng term 1]
   user@R1# set from protocol direct
   user@R1# set from protocol ripng
   user@R1# set then accept
   ```

5. Apply the routing policy.

   In Junos OS, you can only apply RIPng export policies at the group level.

   ```
   [edit protocols ripng group ripng-group]
   user@R1# set export advertise-routes-through-ripng
   ```

Results

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

```
user@R1# show interfaces
fe-1/2/0 {
    unit 1 {
        description to-R2;
        family inet6 {
            address 2001:db8:0:1::/64 {
                eui-64;
            }
        }
    }
}
lo0 {
    unit 1 {
        family inet6 {
            address 2001:db8::1/128;
        }
    }
}
```

```
user@R1# show protocols
ripng {
    traceoptions {
        file ripng-trace-file;
        flag route;
    }
    group ripng-group {
        export advertise-routes-through-ripng;
        neighbor fe-1/2/0.1;
    }
}
```

```
user@R1# show policy-options
policy-statement advertise-routes-through-ripng {
```

```
    term 1 {
        from protocol [ direct ripng ];
        then accept;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Checking the Log File

Purpose

Make sure that the RIPng route updates are logged in the configured log file.

Action

1. Deactivate the extra loopback interface address on Device R3.

```
[edit  interfaces lo0 unit 3 family inet6]
user@R3# deactivate address 2001:db8::3/128
user@R3# commit
```

2. From operational mode, enter the `show log ripng-trace-file` command with the **| match 2001:db8::3** option.

```
user@R1> show log ripng-trace-file | match 2001:db8::3

Mar  6 14:57:03.516867  2001:db8::3/128: metric-in: 3, change: 3 -> 3; # gw: 1, pkt_upd_src
fe80::2a0:a514:0:24c, inx: 0, rte_upd_src fe80::2a0:a514:0:24c
Mar  6 14:57:32.786286  2001:db8::3/128: metric-in: 3, change: 3 -> 3; # gw: 1, pkt_upd_src
```

```
fe80::2a0:a514:0:24c, inx: 0, rte_upd_src fe80::2a0:a514:0:24c
Mar  6 14:58:02.584669  2001:db8::3/128: metric-in: 3, change: 3 -> 3; # gw: 1, pkt_upd_src
fe80::2a0:a514:0:24c, inx: 0, rte_upd_src fe80::2a0:a514:0:24c
Mar  6 14:58:30.213894  2001:db8::3/128: metric-in: 3, change: 3 -> 3; # gw: 1, pkt_upd_src
fe80::2a0:a514:0:24c, inx: 0, rte_upd_src fe80::2a0:a514:0:24c
Mar  6 14:59:00.115110  2001:db8::3/128: metric-in: 3, change: 3 -> 3; # gw: 1, pkt_upd_src
fe80::2a0:a514:0:24c, inx: 0, rte_upd_src fe80::2a0:a514:0:24c
Mar  6 14:59:05.826644 Setting RIPng rtbit on route 2001:db8::3/128, tsi = 0xbb69880
Mar  6 14:59:13.014652  2001:db8::3/128: metric-in: 16, change: 3 -> 16; # gw: 1, pkt_upd_src
fe80::2a0:a514:0:24c, inx: 0, rte_upd_src fe80::2a0:a514:0:24c
Mar  6 14:59:13.015132 CHANGE   2001:db8::3/128     nhid 566 gw fe80::2a0:a514:0:24c RIPng
pref 100/0 metric 3/0 fe-1/2/0.1 **Delete Int>
Mar  6 14:59:13.015197 Best route to 2001:db8::3/128 got deleted. Doing route calculation on
the stored rte-info
```

Meaning

The output shows that the route to 2001:db8::3/128 was deleted.

4

CHAPTER

Troubleshooting

Troubleshooting Network Issues

IN THIS SECTION

Working with Problems on Your Network

IN THIS SECTION

Problem

Description

This checklist provides links to troubleshooting basics, an example network, and includes a summary of the commands you might use to diagnose problems with the router and network.

Solution

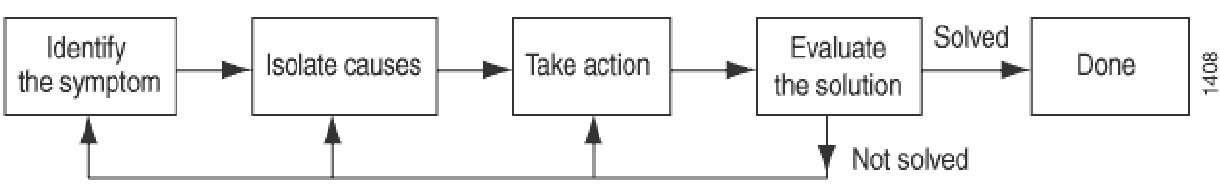
Table 3: Checklist for Working with Problems on Your Network

Tasks	Command or Action
Isolating a Broken Network Connection	
1. *Identifying the Symptoms of a Broken Network Connection*	**ping (*ip-address* | *hostname*) show route (*ip-address* | *hostname*) traceroute (*ip-address* | *hostname*)**
1. *Isolating the Causes of a Network Problem*	`show < configuration | interfaces | protocols | route >`
1. *Taking Appropriate Action for Resolving the Network Problem*	`[edit] delete routing options static route` *`destination-prefix`* **commit and-quit show route *destination-prefix***
1. *Evaluating the Solution to Check Whether the Network Problem Is Resolved*	`show route (`*`ip-address`*` | `*`hostname`*`) ping (`*`ip-address`*` | `*`hostname`*`)` **count 3 traceroute (*ip-address* | *hostname*)**

Isolating a Broken Network Connection

By applying the standard four-step process illustrated in Figure 24 on page 194, you can isolate a failed node in the network. Note that the functionality described in this section is not supported in versions 15.1X49, 15.1X49-D30, or 15.1X49-D40.

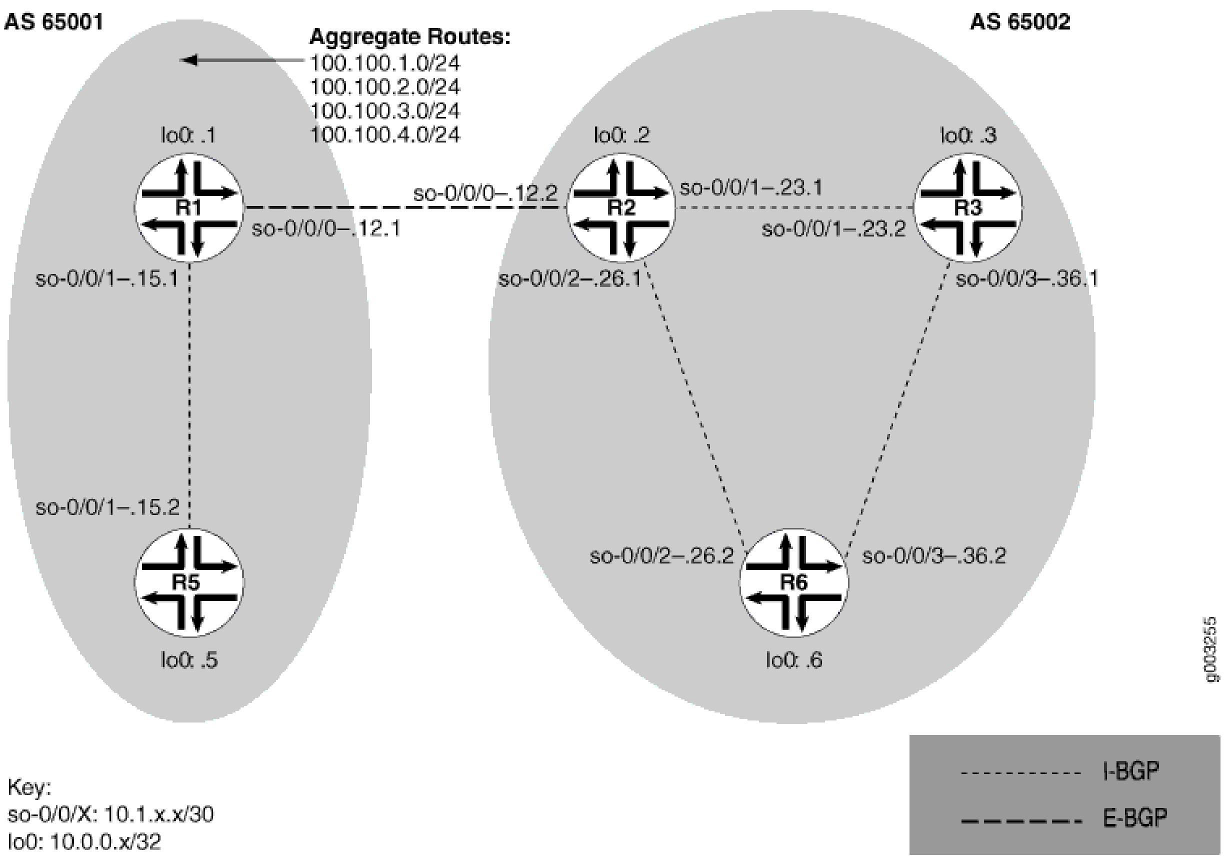
Figure 24: Process for Diagnosing Problems in Your Network



Before you embark on the four-step process, however, it is important that you are prepared for the inevitable problems that occur on all networks. While you might find a solution to a problem by simply trying a variety of actions, you can reach an appropriate solution more quickly if you are systematic in your approach to the maintenance and monitoring of your network. To prepare for problems on your network, understand how the network functions under normal conditions, have records of baseline network activity, and carefully observe the behavior of your network during a problem situation.

Figure 25 on page 195 shows the network topology used in this topic to illustrate the process of diagnosing problems in a network.

Figure 25: Network with a Problem



The network in Figure 25 on page 195 consists of two autonomous systems (ASs). AS 65001 includes two routers, and AS 65002 includes three routers. The border router (`R1`) in AS 65001 announces aggregated prefixes `100.100/24` to the AS 65002 network. The problem in this network is that `R6` does not have access to `R5` because of a loop between `R2` and `R6`.

To isolate a failed connection in your network, follow the steps in these topics:

- *Isolating the Causes of a Network Problem*

- *Taking Appropriate Action for Resolving the Network Problem*
- *Taking Appropriate Action for Resolving the Network Problem*
- *Evaluating the Solution to Check Whether the Network Problem Is Resolved*

Identifying the Symptoms of a Broken Network Connection

IN THIS SECTION

Problem

Description

The symptoms of a problem in your network are usually quite obvious, such as the failure to reach a remote host.

Solution

To identify the symptoms of a problem on your network, start at one end of your network and follow the routes to the other end, entering all or one of the following Junos OS command-line interfaces (CLI) operational mode commands:

```
            user@host> ping (ip-address | host-name)
user@host> show route (ip-address | host-name)
user@host> traceroute (ip-address | host-name)
```

Sample Output

```
user@R6> ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS  Len   ID Flg  off TTL Pro  cks      Src      Dst
```

```
 4  5  00 0054 e2db   0 0000  01  01 a8c6 10.1.26.2  10.0.0.5

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS  Len   ID Flg  off TTL Pro  cks      Src      Dst
 4  5  00 0054 e2de   0 0000  01  01 a8c3 10.1.26.2  10.0.0.5

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS  Len   ID Flg  off TTL Pro  cks      Src      Dst
 4  5  00 0054 e2e2   0 0000  01  01 a8bf 10.1.26.2  10.0.0.5

^C
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 0 packets received, 100% packet loss

user@R6> show route 10.0.0.5

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32        *[IS-IS/165] 00:02:39, metric 10
                     > to 10.1.26.1  via so-0/0/2.0

user@R6> traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 40 byte packets
 1  10.1.26.1 (10.1.26.1)  0.649 ms  0.521 ms  0.490 ms
 2  10.1.26.2 (10.1.26.2)  0.521 ms  0.537 ms  0.507 ms
 3  10.1.26.1 (10.1.26.1)  0.523 ms  0.536 ms  0.514 ms
 4  10.1.26.2 (10.1.26.2)  0.528 ms  0.551 ms  0.523 ms
 5  10.1.26.1 (10.1.26.1)  0.531 ms  0.550 ms  0.524 ms
```

Meaning

The sample output shows an unsuccessful `ping` command in which the packets are being rejected because the time to live is exceeded. The output for the `show route` command shows the interface (`10.1.26.1`) that you can examine further for possible problems. The `traceroute` command shows the loop between `10.1.26.1` (`R2`) and `10.1.26.2` (`R6`), as indicated by the continuous repetition of the two interface addresses.

Isolating the Causes of a Network Problem

IN THIS SECTION

Problem

Description

A particular symptom can be the result of one or more causes. Narrow down the focus of your search to find each individual cause of the unwanted behavior.

Solution

To isolate the cause of a particular problem, enter one or all of the following Junos OS CLI operational mode command:

```
               user@host> show < configuration | bgp | interfaces | isis | ospf | route
>
```

Your particular problem may require the use of more than just the commands listed above. See the appropriate command reference for a more exhaustive list of commonly used operational mode commands.

Sample Output

```
user@R6> show interfaces terse
Interface               Admin Link Proto Local                 Remote
so-0/0/0                up    up
so-0/0/0.0              up    up   inet  10.1.56.2/30
                                   iso
so-0/0/2                up    up
so-0/0/2.0              up    up   inet  10.1.26.2/30
                                   iso
so-0/0/3                up    up
```

```
so-0/0/3.0              up    up   inet  10.1.36.2/30
                                   iso
[...Output truncated...]
```

The following sample output is from R2:

```
user@R2> show route 10.0.0.5

inet.0: 22 destinations, 25 routes (22 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32         *[Static/5] 00:16:21
                    > to 10.1.26.2 via so-0/0/2.0
                    [BGP/170] 3d 20:23:35, MED 5, localpref 100
                      AS path: 65001 I
                    > to 10.1.12.1 via so-0/0/0.0
```

Meaning

The sample output shows that all interfaces on R6 are up. The output from R2 shows that a static route [Static/5] configured on R2 points to R6 (10.1.26.2) and is the preferred route to R5 because of its low preference value. However, the route is looping from R2 to R6, as indicated by the missing reference to R5 (10.1.15.2).

Taking Appropriate Action for Resolving the Network Problem

IN THIS SECTION

Problem

Description

The appropriate action depends on the type of problem you have isolated. In this example, a static route configured on R2 is deleted from the [routing-options] hierarchy level. Other appropriate actions might include the following:

Solution

- Check the local router's configuration and edit it if appropriate.
- Troubleshoot the intermediate router.
- Check the remote host configuration and edit it if appropriate.
- Troubleshoot routing protocols.
- Identify additional possible causes.

To resolve the problem in this example, enter the following Junos OS CLI commands:

```
[edit]
                          user@R2# delete routing-options static route destination-
prefix
user@R2# commit and-quit
user@R2# show route destination-prefix
```

Sample Output

```
[edit]
user@R2# delete routing-options static route 10.0.0.5/32

[edit]
user@R2# commit and-quit
commit complete
Exiting configuration mode

user@R2> show route 10.0.0.5

inet.0: 22 destinations, 24 routes (22 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
10.0.0.5/32        *[BGP/170] 3d 20:26:17, MED 5, localpref 100
                      AS path: 65001 I
                    > to 10.1.12.1 via so-0/0/0.0
```

Meaning

The sample output shows the static route deleted from the [routing-options] hierarchy and the new configuration committed. The output for the show route command now shows the BGP route as the preferred route, as indicated by the asterisk (*).

Evaluating the Solution to Check Whether the Network Problem Is Resolved

IN THIS SECTION

Problem

Description

If the problem is solved, you are finished. If the problem remains or a new problem is identified, start the process over again.

You can address possible causes in any order. In relation to the network in *Isolating a Broken Network Connection*, we chose to work from the local router toward the remote router, but you might start at a different point, particularly if you have reason to believe that the problem is related to a known issue, such as a recent change in configuration.

Solution

To evaluate the solution, enter the following Junos OS CLI commands:

```
user@host> show route (ip-address   |host-name)
user@host> ping (ip-address  | host-name)
user@host> traceroute (ip-address  | host-name)
```

Sample Output

```
user@R6> show route 10.0.0.5

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32        *[BGP/170]  00:01:35, MED 5, localpref 100, from 10.0.0.2
                     AS path: 65001 I
                   > to 10.1.26.1 via so-0/0/2.0

user@R6> ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
64 bytes from 10.0.0.5: icmp_seq=0 ttl=253 time=0.866 ms
64 bytes from 10.0.0.5: icmp_seq=1 ttl=253 time=0.837 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=253 time=0.796 ms
^C
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.796/0.833/0.866/0.029 ms

user@R6> traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 40 byte packets
 1  10.1.26.1 (10.1.26.1)  0.629 ms  0.538 ms  0.497 ms
 2  10.1.12.1 (10.1.12.1)  0.534 ms  0.538 ms  0.510 ms
 3  10.0.0.5 (10.0.0.5)  0.776 ms  0.705 ms  0.672 ms
```

Meaning

The sample output shows that there is now a connection between `R6` and `R5`. The `show route` command shows that the BGP route to `R5` is preferred, as indicated by the asterisk (`*`). The `ping` command is successful and the `traceroute` command shows that the path from `R6` to `R5` is through `R2` (`10.1.26.1`), and then through `R1` (`10.1.12.1`).

Checklist for Tracking Error Conditions

IN THIS SECTION

Problem

Description

Table 4 on page 203 provides links and commands for configuring routing protocol daemon tracing, Border Gateway Protocol (BGP), Intermediate System-to-Intermediate System (IS-IS) protocol, and Open Shortest Path First (OSPF) protocol tracing to diagnose error conditions.

Solution

Table 4: Checklist for Tracking Error Conditions

Tasks	Command or Action
Configure Routing Protocol Process Tracing	
1. *Configure Routing Protocol Process Tracing*	`[edit] edit routing-options traceoptions` *`filename`* `size` *`size`* files ***number*** show co log ***filename***
1. *Configure Routing Protocol Tracing for a Specific Routing Protocol*	`[edit] edit protocol` *`protocol-name`* trace ***filename*** size ***size*** files ***number*** show com log ***filename***
1. *Monitor Trace File Messages Written in Near-Real Time*	monitor start ***filename***
1. *Stop Trace File Monitoring*	monitor stop ***filename***

Table 4: Checklist for Tracking Error Conditions *(Continued)*

Tasks	Command or Action
Configure BGP-Specific Options	
1. Display Detailed BGP Protocol Information	`[edit] edit protocol bgp traceoptions se` `detail show commit run show log` *`filename`*
1. Display Sent or Received BGP Packets	`[edit] edit protocol bgp traceoptions se` `(send | receive) show commit run show lo`
1. Diagnose BGP Session Establishment Problems	`[edit] edit protocol bgp set traceoption` `detail show commit run show log` *`filename`*
Configure IS-IS-Specific Options	
1. Displaying Detailed IS-IS Protocol Information	`[edit] edit protocol isis traceoptions s` `detail show commit run show log` *`filename`*
1. Displaying Sent or Received IS-IS Protocol Packets	`[edit] edit protocols isis traceoptions` `(send | receive) show commit run show lo`
1. Analyzing IS-IS Link-State PDUs in Detail	`[edit] edit protocols isis traceoptions` `detail show commit run show log` *`filename`*
Configure OSPF-Specific Options	
1. Diagnose OSPF Session Establishment Problems	`[edit] edit protocols ospf traceoptions` `detail show commit run show log` *`filename`*
1. Analyze OSPF Link-State Advertisement Packets in Detail	`[edit] edit protocols ospf traceoptions` `update detail show commit run show log`

Configure Routing Protocol Process Tracing

IN THIS SECTION

Action

To configure routing protocol process (rpd) tracing, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit routing-options traceoptions
```

2. Configure the file, file size, number, and flags:

```
[edit routing-options traceoptions]
user@host#  set file filename size size file number
[edit routing-options traceoptions]
user@host#  set flag flag
```

   For example:

```
[edit routing-options traceoptions]
user@host# set file daemonlog size 10240 files 10
[edit routing-options traceoptions]
user@host# set flag general
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit routing-options traceoptions]
user@host# show
file daemonlog size 10k files 10;
flag general;
```

4. Commit the configuration:

```
user@host# commit
```

NOTE: Some traceoptions flags generate an extensive amount of information. Tracing can also slow down the operation of routing protocols. Delete the traceoptions configuration if you no longer require it.

1. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
[edit routing-options traceoptions]
user@pro4-a# run show log daemonlog
Sep 17 14:17:31 trace_on: Tracing to "/var/log/daemonlog" started
Sep 17 14:17:31 Tracing flags enabled: general
Sep 17 14:17:31 inet_routerid_notify: Router ID: 10.255.245.44
Sep 17 14:17:31 inet_routerid_notify: No Router ID assigned
Sep 17 14:17:31 Initializing LSI globals
Sep 17 14:17:31 LSI initialization complete
Sep 17 14:17:31 Initializing OSPF instances
Sep 17 14:17:31 Reinitializing OSPFv2 instance master
Sep 17 14:17:31 OSPFv2 instance master running
[...Output truncated...]
```

Meaning

Table 5 on page 207 lists tracing flags and example output for Junos-supported routing protocol daemon tracing.

Table 5: Routing Protocol Daemon Tracing Flags

Tracing Flag	Description	Example Output
all	All operations	Not available.
general	Normal operations and routing table change	Not available.
normal	Normal operations	Not available.
policy	Policy operations and actions	Nov 29 22:19:58 export: Dest 10.0.0.0 proto Static Nov 29 22:19:58 policy_match_qual_or: Qualifier proto Sense: 0 Nov 29 22:19:58 policy_match_qual_or: Qualifier proto Sense: 0 Nov 29 22:19:58 export: Dest 10.10.10.0 proto IS-IS
route	Routing table changes	Nov 29 22:23:59 Nov 29 22:23:59 rtlist_walker_job: rt_list walk for RIB inet.0 started with 42 entries Nov 29 22:23:59 rt_flash_update_callback: flash KRT (inet.0) start Nov 29 22:23:59 rt_flash_update_callback: flash KRT (inet.0) done Nov 29 22:23:59 rtlist_walker_job: rt_list walk for inet.0 ended with 42 entries Nov 29 22:23:59 Nov 29 22:23:59 KRT Request: send len 68 v14 seq 0 CHANGE route/user af 2 addr 172.16.0.0 nhop-type unicast nhop 10.10.10.33 Nov 29 22:23:59 KRT Request: send len 68 v14 seq 0 ADD route/user af 2 addr 172.17.0.0 nhop-type unicast nhop 10.10.10.33 Nov 29 22:23:59 KRT Request: send len 68 v14 seq 0 ADD route/user af 2 addr 10.149.3.0 nhop-type unicast nhop 10.10.10.33 Nov 29 22:24:19 trace_on: Tracing to "/var/log/rpdlog" started Nov 29 22:24:19 KRT Request: send len 68 v14 seq 0 DELETE route/user af 2 addr 10.10.218.0 nhop-type unicast nhop 10.10.10.29 Nov 29 22:24:19 RELEASE 10.10.218.0 255.255.255.0 gw 10.10.10.29,10.10.10.33 BGP pref 170/-101 metric so-1/1/0.0,so-1/1/1.0 <Release Delete Int Ext> as 65401 Nov 29 22:24:19 KRT Request: send len 68 v14 seq 0 DELETE route/user af 2 addr 172.18.0.0 nhop-type unicast nhop 10.10.10.33
state	State transitions	Not available.

Table 5: Routing Protocol Daemon Tracing Flags *(Continued)*

Tracing Flag	Description	Example Output
task	Interface transactions and processing	Nov 29 22:50:04 foreground dispatch running job task_collect for task Scheduler Nov 29 22:50:04 task_collect_job: freeing task MGMT_Listen (DELETED) Nov 29 22:50:04 foreground dispatch completed job task_collect for task Scheduler Nov 29 22:50:04 background dispatch running job rt_static_update for task RT Nov 29 22:50:04 task_job_delete: delete background job rt_static_update for task RT Nov 29 22:50:04 background dispatch completed job rt_static_update for task RT Nov 29 22:50:04 background dispatch running job Flash update for task RT Nov 29 22:50:04 background dispatch returned job Flash update for task RT Nov 29 22:50:04 background dispatch running job Flash update for task RT Nov 29 22:50:04 task_job_delete: delete background job Flash update for task RT Nov 29 22:50:04 background dispatch completed job Flash update for task RT Nov 29 22:50:04 background dispatch running job Flash update for task RT Nov 29 22:50:04 task_job_delete: delete background job Flash update for task RT
timer	Timer usage	Nov 29 22:52:07 task_timer_hiprio_dispatch: ran 1 timer Nov 29 22:52:07 main: running normal priority timer queue Nov 29 22:52:07 main: ran 1 timer Nov 29 22:52:07 task_timer_hiprio_dispatch: running high priority timer queue Nov 29 22:52:07 task_timer_hiprio_dispatch: ran 1 timer Nov 29 22:52:07 main: running normal priority timer queue Nov 29 22:52:07 main: ran 1 timer Nov 29 22:52:07 main: running normal priority timer queue Nov 29 22:52:07 main: ran 2 timers

Configure Routing Protocol Tracing for a Specific Routing Protocol

IN THIS SECTION

Action

To configure routing protocol tracing for a specific routing protocol, follow these steps:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
                 user@host# edit protocol protocol-name traceoptions
```

2. Configure the file, file size, number, and flags:

```
[edit protocols protocol name traceoptions]
                 user@host# set file filename size size files
number
[edit protocols protocol name traceoptions]
user@host# set flag flag
```

For example:

```
[edit protocols ospf traceoptions]
user@host# set file ospflog size 10240 files 10
[edit protocols ospf traceoptions]
user@host# set flag general
```

3. Verify the configuration:

```
user@host# show
```

For example:

```
[edit protocols ospf traceoptions]
user@host# show
file ospflog size 10k files 10;
flag general;
```

4. Commit the configuration:

```
user@host# commit
```

5. View the contents of the file containing the detailed messages:

```
user@host# run show log filename
```

For example:

```
[edit protocols ospf traceoptions]
user@pro4-a# run show log ospflog
Sep 17 14:23:10 trace_on: Tracing to "/var/log/ospflog" started
Sep 17 14:23:10 rt_flash_update_callback: flash OSPF (inet.0) start
Sep 17 14:23:10 OSPF:  multicast address 224.0.0.5/32, route ignored
Sep 17 14:23:10 rt_flash_update_callback: flash OSPF (inet.0) done
Sep 17 14:23:10 CHANGE   10.255.245.46/32   gw 10.10.208.67  OSPF     pref 10/0 metric 1/0
fe-0/0/0.0 <Delete Int>
Sep 17 14:23:10 CHANGE   10.255.245.46/32   gw 10.10.208.67  OSPF     pref 10/0 metric 1/0
fe-0/0/0.0 <Active Int>
Sep 17 14:23:10 ADD      10.255.245.46/32   gw 10.10.208.67  OSPF     pref 10/0 metric 1/0
fe-0/0/0.0 <Active Int>
Sep 17 14:23:10 CHANGE   10.255.245.48/32   gw 10.10.208.69  OSPF     pref 10/0 metric 1/0
fe-0/0/0.0 <Delete Int>
Sep 17 14:23:10 CHANGE   10.255.245.48/32   gw 10.10.208.69  OSPF     pref 10/0 metric 1/0
fe-0/0/0.0 <Active Int>
Sep 17 14:23:10 ADD      10.255.245.48/32   gw 10.10.208.69  OSPF     pref 10/0 metric 1/0
fe-0/0/0.0 <Active Int>
Sep 17 14:23:10 rt_close: 4/4 routes proto OSPF
[...Output truncated...]
```

Meaning

Table 6 on page 210 lists standard tracing options that are available globally or that can be applied to specific protocols. You can also configure tracing for a specific BGP peer or peer group. For more information, see the *Junos System Basics Configuration Guide*.

Table 6: Standard Trace Options for Routing Protocols

Tracing Flag	Description
all	All operations

Table 6: Standard Trace Options for Routing Protocols *(Continued)*

Tracing Flag	Description
general	Normal operations and routing table changes
normal	Normal operations
policy	Policy operations and actions
route	Routing table changes
state	State transitions
task	Interface transactions and processing
timer	Timer usage

Monitor Trace File Messages Written in Near-Real Time

IN THIS SECTION

Purpose

To monitor messages in near-real time as they are being written to a trace file.

Action

To monitor messages in near-real time as they are being written to a trace file, use the following Junos OS command-line interface (CLI) operational mode command:

```
user@host> monitor start  filename
```

Sample Output

command-name

```
user@host> monitor start isis
user@host>
*** isis ***
Sep 15 18:32:21 Updating LSP isis5.02-00 in database
Sep 15 18:32:21 Updating L2 LSP isis5.02-00 in TED
Sep 15 18:32:21 Adding a half link from isis5.02 to isis6.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis5.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis6.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis5.00
Sep 15 18:32:21 Scheduling L2 LSP isis5.02-00 sequence 0xd87 on interface fxp2.3
Sep 15 18:32:21 Updating LSP isis5.00-00 in database
Sep 15 18:32:21 Updating L1 LSP isis5.00-00 in TED
Sep 15 18:32:21 Sending L2 LSP isis5.02-00 on interface fxp2.3
Sep 15 18:32:21     sequence 0xd87, checksum 0xc1c8, lifetime 1200
```

Stop Trace File Monitoring

IN THIS SECTION

Action

To stop monitoring a trace file in near-real time, use the following Junos OS CLI operational mode command after you have started monitoring:

```
user@host              monitor stop filename
```

Sample Output

```
user@host> monitor start isis
user@host>
*** isis ***
Sep 15 18:32:21 Updating LSP isis5.02-00 in database
Sep 15 18:32:21 Updating L2 LSP isis5.02-00 in TED
Sep 15 18:32:21 Adding a half link from isis5.02 to isis6.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis5.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis6.00
Sep 15 18:32:21 Adding a half link from isis5.02 to isis5.00
Sep 15 18:32:21 Scheduling L2 LSP isis5.02-00 sequence 0xd87 on interface fxp2.3
Sep 15 18:32:21 Updating LSP isis5.00-00 in database
Sep 15 18:32:21 Updating L1 LSP isis5.00-00 in TED
Sep 15 18:32:21 Sending L2 LSP isis5.02-00 on interface fxp2.3
Sep 15 18:32:21     sequence 0xd87, checksum 0xc1c8, lifetime 1200
monitor stop isis
user@host>
```

Monitoring RIP Traffic

IN THIS SECTION

Monitoring RIP Routing Information

IN THIS SECTION

Purpose

NOTE: This topic applies only to the J-Web Application package.

Use the monitoring functionality to monitor RIP routing on routing devices.

Action

To view RIP routing information in the J-Web interface, select **Monitor** > **Routing** > **RIP Information**.

To view RIP routing information in the CLI, enter the following CLI commands:

- `show rip statistics`
- `show rip neighbor`

Meaning

Table 7 on page 214 summarizes key output fields in the RIP routing display in the J-Web interface.

Table 7: Summary of Key RIP Routing Output Fields

Field	Values	Additional Information
RIP Statistics		
Protocol Name	The RIP protocol name.	

Table 7: Summary of Key RIP Routing Output Fields *(Continued)*

Field	Values	Additional Information
Port number	The port on which RIP is enabled.	
Hold down time	The interval during which routes are neither advertised nor updated.	
Global routes learned	Number of RIP routes learned on the logical interface.	
Global routes held down	Number of RIP routes that are not advertised or updated during the hold-down interval.	
Global request dropped	Number of requests dropped.	
Global responses dropped	Number of responses dropped.	
RIP Neighbors		
Neighbor	Name of the RIP neighbor.	This value is the name of the interface on which RIP is enabled. Click the name to see the details for this neighbor.
State	State of the RIP connection: **Up** or **Dn** (Down).	
Source Address	Local source address.	This value is the configured address of the interface on which RIP is enabled.
Destination Address	Destination address.	This value is the configured address of the immediate RIP adjacency.
Send Mode	The mode of sending RIP messages.	

Table 7: Summary of Key RIP Routing Output Fields *(Continued)*

Field	Values	Additional Information
Receive Mode	The mode in which messages are received.	
In Metric	Value of the incoming metric configured for the RIP neighbor.	

Verifying a RIP Configuration

IN THIS SECTION

To verify a RIP configuration, perform the following tasks:

Verifying the RIP-Enabled Interfaces

IN THIS SECTION

Purpose

Verify that all the RIP-enabled interfaces are available and active.

Action

From the CLI, enter the `show rip neighbor` command.

Sample Output

command-name

```
user@host> show rip neighbor
Source          Destination     Send   Receive   In
Neighbor          State  Address         Address         Mode   Mode     Met
--------          -----  -------         -----------     ----   -------  ---
ge-0/0/0.0           Dn (null)          (null)          mcast  both       1
ge-0/0/1.0           Up 192.168.220.5   224.0.0.9       mcast  both       1
```

Meaning

The output shows a list of the RIP neighbors that are configured on the device. Verify the following information:

- Each configured interface is present. Interfaces are listed in alphabetical order.
- Each configured interface is up. The state of the interface is listed in the **Destination State** column. A state of **Up** indicates that the link is passing RIP traffic. A state of **Dn** indicates that the link is not passing RIP traffic. In a point-to-point link, this state generally means that either the end point is not configured for RIP or the link is unavailable.

Verifying Reachability of All Hosts in the RIP Network

IN THIS SECTION

Purpose

By using the traceroute tool on each loopback address in the network, verify that all hosts in the RIP network are reachable from each Juniper Networks device.

Action

For each device in the RIP network:

1. In the J-Web interface, select **Troubleshoot>Traceroute**.
2. In the Remote Host box, type the name of a host for which you want to verify reachability from the device.
3. Click **Start**. Output appears on a separate page.

Sample Output

command-name

```
1 172.17.40.254 (172.17.40.254) 0.362 ms 0.284 ms 0.251 ms
2 routera-fxp0.englab.mycompany.net (192.168.71.246) 0.251 ms 0.235 ms 0.200 ms
```

Meaning

Each numbered row in the output indicates a routing hop in the path to the host. The three-time increments indicate the round-trip time (RTT) between the device and the hop for each traceroute packet.

To ensure that the RIP network is healthy, verify the following information:

- The final hop in the list is the host you want to reach.
- The number of expected hops to the host matches the number of hops in the traceroute output. The appearance of more hops than expected in the output indicates that a network segment is probably unreachable. It might also indicate that the incoming or outgoing metric on one or more hosts has been set unexpectedly.

Verifying the Exchange of RIP Messages

IN THIS SECTION

Purpose

Verify that RIP messages are being sent and received on all RIP-enabled interfaces.

Action

From the CLI, enter the `show rip statistics` command.

Sample Output

command-name

```
user@host> show rip statistics
RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
             10              0              0              0

t1-0/0/2.0:  0 routes learned; 13 routes advertised; timeout 120s; update interval 45s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                     2855           11            2
Triggered Updates Sent              5            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received             41            0            0
RIPv2 Bad Route Entries             0            0            0
```

```
RIPv2 Updates Ignored               0           0           0
Authentication Failures             0           0           0
RIP Requests Received               0           0           0
RIP Requests Ignored                0           0           0

ge-0/0/1.0:  10 routes learned; 3 routes advertised; timeout 180s; update interval 30s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                     2855          11            2
Triggered Updates Sent              3           0            0
Responses Sent                      0           0            0
Bad Messages                        1           0            0
RIPv1 Updates Received              0           0            0
RIPv1 Bad Route Entries             0           0            0
RIPv1 Updates Ignored               0           0            0
RIPv2 Updates Received           2864          11            2
RIPv2 Bad Route Entries            14           0            0
RIPv2 Updates Ignored               0           0            0
Authentication Failures             0           0            0
RIP Requests Received               0           0            0
RIP Requests Ignored                0           0            0
```

Meaning

The output shows the number of RIP routes learned. It also shows the number of RIP updates sent and received on the RIP-enabled interfaces. Verify the following information:

- The number of RIP routes learned matches the number of expected routes learned. Subnets learned by direct connectivity through an outgoing interface are not listed as RIP routes.
- RIP updates are being sent on each RIP-enabled interface. If no updates are being sent, the routing policy might not be configured to export routes.
- RIP updates are being received on each RIP-enabled interface. If no updates are being received, the routing policy might not be configured to export routes on the host connected to that subnet. The lack of updates might also indicate an authentication error.

RELATED DOCUMENTATION

RIP Configuration Overview | 12

show rip statistics | 348

5

CHAPTER

RIP Configuration Statements

any-sender

IN THIS SECTION

Syntax

```
any-sender;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Disable strict sender address checks.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 8.0.

authentication-key (Protocols RIP)

IN THIS SECTION

Syntax

```
authentication-key password;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Require authentication for RIP route queries received on an interface.

Options

password—Authentication password. If the password does not match, the packet is rejected. The password can be from 1 through 16 contiguous characters long and can include any ASCII strings.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Configuring Route Authentication for RIP using single MD5 key | 25

authentication-type (Protocols RIP)

IN THIS SECTION

Syntax

```
authentication-type type;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
```

```
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Configure the type of authentication for RIP route queries received on an interface.

Default

If you do not include this statement and the `authentication-key` statement, RIP authentication is disabled.

Options

type—Authentication type:

- `md5`—Use the MD5 algorithm to create an encoded checksum of the packet. The encoded checksum is included in the transmitted packet. The receiving routing device uses the authentication key to verify the packet, discarding it if the digest does not match. This algorithm provides a more secure authentication scheme.
- `none`—Disable authentication. If `none` is configured, the configured authentication key is ignored.
- `simple`—Use a simple password. The password is included in the transmitted packet, which makes this method of authentication relatively insecure. The password can be from 1 through 16 contiguous letters or digits long.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Configuring Route Authentication for RIP using single MD5 key | 25

authentication-key (Protocols RIP) | 225

bfd-liveness-detection (Protocols RIP)

IN THIS SECTION

Syntax

```
bfd-liveness-detection {
    authentication {
        algorithm algorithm-name;
        key-chain key-chain-name;
        loose-check;
    }
    detection-time {
        threshold milliseconds;
    }
```

```
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        multiplier number;
        no-adaptation;
        transmit-interval {
            minimum-interval milliseconds;
            threshold milliseconds;
        }
        version (1 | automatic);
    }
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip group group-name],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Configure bidirectional failure detection timers and authentication.

The remaining statements are explained separately. See CLI Explorer.

Options

`authentication algorithm` *`algorithm-name`* —Configure the algorithm used to authenticate the specified BFD session: `simple-password`, `keyed-md5`, `keyed-sha-1`, `meticulous-keyed-md5`, or `meticulous-keyed-sha-1`.

`authentication key-chain` *`key-chain-name`*—Associate a security key with the specified BFD session using the name of the security keychain. The name you specify must match one of the keychains configured in the `authentication-key-chains key-chain` statement at the `[edit security]` hierarchy level.

`authentication loose-check`—(Optional) Configure loose authentication checking on the BFD session. Use only for transitional periods when authentication is not configured at both ends of the BFD session.

`detection-time threshold` *`milliseconds`*—Configure a threshold for the adaptation of the BFD session detection time. When the detection time adapts to a value equal to or greater than the threshold, a single trap and a single system log message are sent.

`minimum-interval` *`milliseconds`*—Configure the minimum interval after which the local routing device transmits a hello packet and then expects to receive a reply from the neighbor with which it has established a BFD session. Optionally, instead of using this statement, you can specify the minimum transmit and receive intervals separately using the `transmit-interval minimum-interval` and `minimum-receive-interval` statements.

- **Range:** 1 through 255,000 milliseconds

`minimum-receive-interval` *`milliseconds`*—Configure the minimum interval after which the local routing device expects to receive a reply from a neighbor with which it has established a BFD session. Optionally, instead of using this statement, you can configure the minimum receive interval using the `minimum-interval` statement.

- **Range:** 1 through 255,000 milliseconds

`multiplier` *`number`*—Configure the number of hello packets not received by a neighbor that causes the originating interface to be declared down.

- **Range:** 1 through 255
- **Default:** 3

`no-adaptation`—Configure BFD sessions not to adapt to changing network conditions. We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

`transmit-interval threshold` *`milliseconds`*—Configure the threshold for the adaptation of the BFD session transmit interval. When the transmit interval adapts to a value greater than the threshold, a single trap and a single system message are sent. The interval threshold must be greater than the minimum transmit interval.

- **Range:** 0 through 4,294,967,295 (2^{32} – 1)

`transmit-interval minimum-interval` *`milliseconds`*—Configure a minimum interval after which the local routing device transmits hello packets to a neighbor. Optionally, instead of using this statement, you can configure the minimum transmit interval using the `minimum-interval` statement.

- **Range:** 1 through 255,000

`version`—Configure the BFD version to detect: `1` (BFD version 1) or `automatic` (autodetect the BFD version).

- **Default:** `automatic`

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 8.0.

Options `detection-time threshold` and `transmit-interval threshold` introduced in Junos OS Release 8.2.

Support for logical systems introduced in Junos OS Release 8.3.

Option `no-adaptation` introduced in Junos OS Release 9.0.

Options `authentication algorithm`, `authentication key-chain`, and `authentication loose-check` introduced in Junos OS Release 9.6.

Options `authentication algorithm`, `authentication key-chain`, and `authentication loose-check` introduced in Junos OS Release 9.6 for EX Series switches.

RELATED DOCUMENTATION

Example: Configuring BFD for RIP | 63
Example: Configuring BFD Authentication for RIP | 74

check-zero

IN THIS SECTION

Syntax

```
(check-zero | no-check-zero);
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Check whether the reserved fields in a RIP packet are zero:

- **check-zero**—Discard version 1 packets that have nonzero values in the reserved fields and version 2 packets that have nonzero values in the fields that must be zero. This default behavior implements the RIP version 1 and version 2 specifications.

- **no-check-zero**—Receive RIP version 1 packets with nonzero values in the reserved fields or RIP version 2 packets with nonzero values in the fields that must be zero. This is in spite of the fact that they are being sent in violation of the specifications in RFC 1058 and RFC 2453.

Default

check-zero

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

demand-circuit (Protocols RIP)

IN THIS SECTION

Syntax

```
demand-circuit;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip group group-name],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip group group-name],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Configure a neighboring interface to act as a RIP demand circuit. To complete the demand circuit, you must configure both ends of the pair as demand circuits. When configured, the device sends RIP information only when changes occur in the routing database.

Default

Disabled. You must explicitly configure two neighboring interfaces to act as a RIP demand circuit.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced in Release 11.1 of Junos OS.

RELATED DOCUMENTATION

Example: Configuring RIP Demand Circuits | 57
RIP Demand Circuits Overview | 54
max-retrans-time | 249

dynamic-peers

IN THIS SECTION

Syntax

```
dynamic-peers;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit protocols rip group group-name neighbor neighbor-name]
```

Description

Configure an interface to have dynamic peers in a point-to-multipoint RIP network.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 11.4.

RELATED DOCUMENTATION

Example: Configuring Point-to-Multipoint RIP Networks | 99

export

IN THIS SECTION

Syntax

```
export [ policy-names ];
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name],
[edit protocols rip group group-name],
[edit routing-instances routing-instance-name protocols rip group group-name]
```

Description

Apply a policy to routes being exported to the neighbors.

By default, RIP does not export routes it has learned to its neighbors. To enable RIP to export routes, apply one or more export policies.

If no routes match the policies, the local routing device does not export any routes to its neighbors. Export policies override any metric values determined through calculations involving the values configured with the `metric-in` and `metric-out` statements.

NOTE: The export policy on RIP does not support manipulating routing information of the next hop.

Options

policy-names—Name of one or more policies.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

import

graceful-restart (Protocols RIP)

IN THIS SECTION

Syntax

```
graceful-restart {
    disable;
    restart-time seconds;
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit protocols rip]
```

Description

Configure graceful restart for RIP.

Options

disable—Disables graceful restart for RIP.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Junos OS High Availability User Guide

group (Protocols RIP)

IN THIS SECTION

Syntax

```
group group-name {
    bfd-liveness-detection {
        authentication {
```

```
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
        version (0 | 1 | automatic);
    }
    demand-circuit;
    export policy;
    max-retrans-time  seconds;
    metric-out metric;
    preference number;
    route-timeout seconds;
    update-interval seconds;
    neighbor neighbor-name {
        authentication-key password;
        authentication-type type;
        bfd-liveness-detection {
            authentication {
                algorithm algorithm-name;
                key-chain key-chain-name;
                loose-check;
            }
            detection-time {
                threshold milliseconds;
            }
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            transmit-interval {
                threshold milliseconds;
                minimum-interval milliseconds;
            }
            multiplier number;
            version (0 | 1 | automatic);
```

```
        }
        (check-zero | no-check-zero);
        demand-circuit;
        import policy-name;
        max-retrans-time seconds;
        message-size number;
        metric-in metric;
        metric-out metric;
        receive receive-options;
        route-timeout seconds;
        send send-options;
        update-interval seconds;
    }
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit protocols rip],
[edit routing-instances routing-instance-name protocols rip]
```

Description

Configure a set of RIP neighbors that share an export policy and metric. The export policy and metric govern what routes to advertise to neighbors in a given group. Each group must contain at least one neighbor. You should create a group for every export policy.

Options

group-name—Name of a group, up to 16 characters long.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Configuring RIP

holddown (Protocols RIP)

IN THIS SECTION

Syntax

```
holddown seconds;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit protocols rip],
[edit routing-instances routing-instance-name protocols rip]
```

Description

Configure how long the expired route is retained in the routing table before being removed.

When the hold-down timer runs on RIP demand circuits, routes are advertised as unreachable on other interfaces. When the hold-down timer expires, the route is removed from the routing table if all destinations detect that the route is unreachable or the remaining destinations are down.

Options

seconds—Estimated time to wait before making updates to the routing table.

- **Range:** 10 through 180 seconds
- **Default:** 180 seconds

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Configuring RIP Timers | 44

RIP Demand Circuits Overview | 54

import (Protocols RIP)

IN THIS SECTION

Syntax

```
import [ policy-names ];
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
```

```
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Apply one or more policies to routes being imported by the local routing device from neighbors.

Options

policy-names—Name of one or more policies.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Applying Policies to RIP Routes Imported from Neighbors | **107**

Junos OS Routing Policies, Firewall Filters, and Traffic Policers User Guide for Routing Devices

export

interface-type (Protocols RIP)

IN THIS SECTION

Syntax

```
interface-type p2mp;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit protocols rip group group-name neighbor neighbor-name]
```

Description

Configure the type of interface in a RIP network.

This statement enables a RIP device to have single or multiple peers through an interface.

Options

p2mp Configure an interface in a RIP network as a point-to-multipoint interface.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 11.4.

RELATED DOCUMENTATION

Example: Configuring Point-to-Multipoint RIP Networks | 99

max-retrans-time

IN THIS SECTION

Syntax

```
max-retrans-time seconds;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip group group-name],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip group group-name],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

RIP demand circuits send update messages every 5 seconds to an unresponsive peer. Configure the retransmission timer to limit the number of times the demand circuit resends update messages to an unresponsive peer. If the configured retransmission threshold is reached, routes from the next hop router are marked as unreachable and the hold-down timer starts. You must configure a pair of RIP demand circuits for this timer to take effect.

To determine the number of times to resend the update message, use the following calculation:

```
5 seconds x number of retransmissions = retransmission seconds
```

Options

seconds—The total amount of time the demand circuit resends update messages to an unresponsive peer. The seconds range corresponds to sending an update message a minimum of 1 time (5 seconds) and a maximum of 36 times (180 seconds).

- **Range:** 5 through 180 seconds
- **Default:** 5 seconds

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 11.1.

RELATED DOCUMENTATION

Example: Configuring RIP Demand Circuits | 57
RIP Demand Circuits Overview | 54
demand-circuit (Protocols RIP) | 234

message-size

IN THIS SECTION

Syntax

```
message-size number;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Specify the number of route entries to be included in every RIP update message. To ensure interoperability with other vendors' equipment, use the standard of 25 route entries per message.

Options

number—Number of route entries per update message.

- **Range:** 25 through 255 entries
- **Default:** 25 entries

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

metric-in (Protocols RIP)

IN THIS SECTION

Syntax

```
metric-in metric;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Specify the metric to add to incoming routes when the routing device advertises into RIP routes that were learned from other protocols. Use this statement to configure the routing device to prefer RIP routes learned through a specific neighbor.

Options

metric—Metric value.

- **Range:** 1 through 16
- **Default:** 1

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Configuring the Metric Value Added to Imported RIP Routes | 90

metric-out

IN THIS SECTION

Syntax

```
metric-out metric;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Specify the metric value to add to routes transmitted to the neighbor. Use this statement to control how other routing devices prefer RIP routes sent from this neighbor.

Options

metric—Metric value.

- **Range:** 1 through 15
- **Default:** 1

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

neighbor

IN THIS SECTION

Syntax

```
neighbor neighbor-name {
    authentication-key password;
    authentication-type type;
    bfd-liveness-detection {
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
        version (0 | 1 | automatic);
    }
    (check-zero | no-check-zero);
```

```
    demand-circuit;
    import policy-name;
    max-retrans-time seconds;
    message-size number;
    metric-in metric;
    metric-out metric;
    receive receive-options;
    route-timeout seconds;
    send send-options;
    update-interval seconds;
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name],
[edit protocols rip group group-name],
[edit routing-instances routing-instance-name protocols rip group group-name]
```

Description

Configure neighbor-specific RIP parameters, thereby overriding the defaults set for the routing device.

Options

neighbor-name—Name of an interface over which a routing device communicates to its neighbors.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

peer (Protocols RIP)

IN THIS SECTION

Syntax

```
peer IP address;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit protocols rip group group-name neighbor neighbor-name]
```

Description

Configure a static peer for an interface in a point-to-multipoint RIP network.

Options

address IP address of the static peer to be configured.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 11.4.

RELATED DOCUMENTATION

Example: Configuring Point-to-Multipoint RIP Networks | 99

preference (Protocols RIP)

IN THIS SECTION

Syntax

```
preference preference;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name],
[edit protocols rip group group-name],
[edit routing-instances routing-instance-name protocols rip group group-name]
```

Description

Specify the preference of external routes learned by RIP as compared to those learned from other routing protocols.

By default, Junos OS assigns a preference of 100 to routes that originate from RIP. When Junos OS determines a route's preference to become the active route, the software selects the route with the lowest preference and installs this route into the forwarding table.

Options

preference—Preference value. A lower value indicates a more preferred route.

- **Range:** 0 through 4,294,967,295 (2^{32} – 1)

- **Default:** 100

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Route Preferences Overview

receive (Protocols RIP)

IN THIS SECTION

Syntax

```
receive receive-options;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Configure RIP receive options.

Options

receive-options—One of the following:

- **both**—Accept both RIP version 1 and version 2 packets.
- **none**—Do not receive RIP packets.
- **version-1**—Accept only RIP version 1 packets.
- **version-2**—Accept only RIP version 2 packets.
- **Default: both**

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Configuring the Sending and Receiving of RIPv1 and RIPv2 Packets | 117
send (Protocols RIP) | 272

rib-group (Protocols RIP)

IN THIS SECTION

Syntax

```
rib-group group-name;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit protocols rip],
[edit routing-instances routing-instance-name protocols rip]
```

Description

Install RIP routes into multiple routing tables by configuring a routing table group.

Options

group-name—Name of the routing table group.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Redistributing Routes Between Two RIP Instances | 124

rip

IN THIS SECTION

Syntax

```
rip {...}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols],
[edit protocols],
[edit routing-instances routing-instance-name protocols]
```

Description

Enable RIP routing on the routing device.

Default

RIP is disabled on the routing device.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Configuring RIP

route-timeout (Protocols RIP)

IN THIS SECTION

Syntax

```
route-timeout seconds;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
[edit routing-instances routing-instance-name protocols rip group group-name],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Configure the route timeout interval for RIP. If a route is not refreshed after being installed in the routing table by the specified timeout interval, the route is marked as invalid and is removed from the routing table after the hold-down period expires.

Options

seconds—Estimated time to wait before making updates to the routing table.

- **Range:** 30 through 360 seconds

- **Default:** 180 seconds

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 7.6.

RELATED DOCUMENTATION

Example: Configuring RIP Timers | 44
RIP Demand Circuits Overview | 54

routing-instances (Multiple Routing Entities)

IN THIS SECTION

Syntax

```
routing-instances routing-instance-name { ... }
```

Hierarchy Level

```
[edit],
[edit logical-systems logical-system-name]
```

Description

Configure an additional routing entity for a router. You can create multiple instances of BGP, IS-IS, OSPF, OSPFv3, and RIP for a router. You can also create multiple routing instances for separating routing tables, routing policies, and interfaces for individual wholesale subscribers (retailers) in a Layer 3 wholesale network.

Each routing instance consist of the following:

- A set of routing tables
- A set of interfaces that belong to these routing tables
- A set of routing option configurations

Each routing instance has a unique name and a corresponding IP unicast table. For example, if you configure a routing instance with the name `my-instance`, its corresponding IP unicast table is my-instance.inet.0. All routes for `my-instance` are installed into my-instance.inet.0.

Routes are installed into the default routing instance inet.0 by default, unless a routing instance is specified.

In Junos OS Release 9.0 and later, you can no longer specify a routing-instance name of *primary*, *default*, or *bgp* or include special characters within the name of a routing instance.

In Junos OS Release 9.6 and later, you can include a slash (/) in a routing-instance name only if a logical system is not configured. That is, you cannot include the slash character in a routing-instance name if a logical system other than the default is explicitly configured. Routing-instance names, further, are restricted from having the form `__.*__` (beginning and ending with underscores). The colon `:` character cannot be used when multitopology routing (MTR) is enabled.

Default

Routing instances are disabled for the router.

Options

`routing-instance-name`	—Name of the routing instance. This must be a non-reserved string of not more than 128 characters.
`remote-vtep-list`	Configure static remote VXLAN tunnel endpoints.
`remote-vtep-v6-list`	Configure static IPv6 remote VXLAN tunnel endpoints.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

`remote-vtep-v6-list` statement introduced in Junos OS Release 17.3 for MX Series routers with MPC and MIC interfaces.

RELATED DOCUMENTATION

Example: Configuring Interprovider Layer 3 VPN Option A

Example: Configuring Interprovider Layer 3 VPN Option B

Example: Configuring Interprovider Layer 3 VPN Option C

send (Protocols RIP)

IN THIS SECTION

Syntax

```
send send-options;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
[edit routing-instances routing-instance-name protocols rip group group-name neighbor neighbor-
name]
```

Description

Configure RIP send options.

Options

send-options—One of the following:

- **broadcast**—Broadcast RIP version 2 packets (RIP version 1 compatible).
- **multicast**—Multicast RIP version 2 packets. This is the default.
- **none**—Do not send RIP updates.
- **version-1**—Broadcast RIP version 1 packets.
- **Default: multicast**

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Configuring the Sending and Receiving of RIPv1 and RIPv2 Packets | **117**

receive (Protocols RIP) | **262**

traceoptions (Protocols RIP)

IN THIS SECTION

Syntax

```
traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit protocols rip],
[edit routing-instances routing-instance-name protocols rip]
```

Description

Set RIP protocol-level tracing options.

> **NOTE**: The `traceoptions` statement is not supported on QFabric systems.

Default

The default RIP protocol-level trace options are inherited from the global `traceoptions` statement.

Options

disable—(Optional) Disable the tracing operation. One use of this option is to disable a single operation when you have defined a broad group of tracing operations, such as **all**.

file ***filename***—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. We recommend that you place RIP tracing output in the file **/var/log/rip-log**.

files ***number***—(Optional) Maximum number of trace files. When a trace file named ***trace-file*** reaches its maximum size, it is renamed ***trace-file*.0**, then ***trace-file*.1**, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum number of files, you must also specify a maximum file size with the **size** option.

- **Range:** 2 through 1000 files
- **Default:** 10 files

flag ***flag***—Tracing operation to perform. To specify more than one tracing operation, include multiple `flag` statements.

RIP Tracing Options

- **auth**—RIP authentication
- **error**—RIP error packets
- **expiration**—RIP route expiration processing
- **holddown**—RIP hold-down processing

- **nsr-synchronization**—Nonstop routing synchronization events
- **packets**—All RIP packets
- **request**—RIP information packets such as request, poll, and poll entry packets
- **trigger**—RIP triggered updates
- **update**—RIP update packets

Global Tracing Options

- **all**—All tracing operations
- **general**—A combination of the **normal** and **route** trace operations
- **normal**—All normal operations
- **Default:** If you do not specify this option, only unusual or abnormal operations are traced.
- **policy**—Policy operations and actions
- **route**—Routing table changes
- **state**—State transitions
- **task**—Routing protocol task processing
- **timer**—Routing protocol timer processing

flag-modifier—(Optional) Modifier for the tracing flag. You can specify one or more of these modifiers:

- **detail**—Provide detailed trace information.
- **receive**—Trace the packets being received.
- **receive-detail**—Provide detailed trace information for packets being received.
- **send**—Trace the packets being transmitted.
- **send-detail**—Provide detailed trace information for packets being transmitted.

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB) or megabytes (MB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum file size, you must also specify a maximum number of trace files with the **files** option.

- **Syntax:** *x***k** to specify KB, *x***m** to specify MB, or *x***g** to specify GB
- **Range:** 10 KB through the maximum file size supported on your system
- **Default:** 128 KB

world-readable—(Optional) Allow any user to read the log file.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Tracing RIP Protocol Traffic

update-interval (Protocols RIP)

IN THIS SECTION

Syntax

```
update-interval seconds;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols rip],
[edit logical-systems logical-system-name protocols rip group group-name],
[edit logical-systems logical-system-name protocols rip group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols rip
group group-name neighbor neighbor-name],
[edit protocols rip],
[edit protocols rip group group-name],
[edit protocols rip  group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols rip],
[edit routing-instances routing-instance-name protocols rip group group-name],
[edit routing-instances routing-instance-name protocols rip  group group-name neighbor neighbor-
name]
```

Description

Configure the interval at which routes learned by RIP are sent to neighbors. This timer controls the interval between routing updates. This timer is set to 30 seconds, by default, with a small random amount of time added when the timer is reset. This added time prevents congestion that can happen if all routing devices update their neighbors simultaneously.

Options

seconds—Estimated time to wait before making updates to the routing table.

- **Range:** 10 through 60 seconds
- **Default:** 30 seconds

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 7.6.

RELATED DOCUMENTATION

Example: Configuring RIP Timers | 44

6

CHAPTER

RIPng Configuration Statements

export (Protocols RIPng)

IN THIS SECTION

Syntax

```
export [ policy-names ];
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng group group-name],
[edit protocols ripng group group-name],
[edit routing-instances routing-instance-name protocols ripng group group-name]
```

Description

Apply a policy or list of policies to routes being exported to the neighbors.

By default, RIPng does not export routes it has learned to its neighbors. To have RIPng export routes, apply one or more export policies. To apply export policies and to filter routes being exported from the local routing device to its neighbors, include the `export` statement and list the name of the policy to be evaluated.

You can define one or more export policies. If no routes match the policies, the local routing device does not export any routes to its neighbors. Export policies override any metric values determined through calculations involving the values configured with the `metric-in` and `metric-out` statements.

Options

policy-names—Name of one or more policies.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

import

graceful-restart (Protocols RIPng)

IN THIS SECTION

Syntax

```
graceful-restart {
    disable;
    restart-time seconds;
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng],
[edit protocols ripng],
[edit routing-instances routing-instance-name protocols ripng]
```

Description

Configure graceful restart for RIPng.

Options

disable—Disables graceful restart for RIPng.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

Junos OS High Availability User Guide

group (Protocols RIPng)

IN THIS SECTION

Syntax

```
group group-name {
    export [ policy-names ];
    metric-out metric;
    neighbor neighbor-name {
        import policy-name;
        metric-in metric;
        receive <none>;
        route-timeout seconds;
        send <none>;
        update-interval seconds;
    }
    preference number;
    route-timeout seconds;
    update-interval seconds;
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng],
[edit protocols ripng],
[edit routing-instances routing-instance-name protocols ripng]
```

Description

Configure a set of RIPng neighbors that share an export policy and metric. The export policy and metric govern what routes to advertise to neighbors in a given group.

Each group must contain at least one neighbor. You should create a group for each export policy that you have.

Options

group-name—Name of a group, up to 16 characters long.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

holddown (Protocols RIPng)

IN THIS SECTION

Syntax

```
holddown seconds;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng],
[edit protocols ripng],
[edit routing-instances routing-instance-name protocols ripng]
```

Description

Configure how long the expired route is retained in the routing table before being removed.

Options

seconds—Estimated time to wait before removing expired routes from the routing table.

- **Default:** 180 seconds
- **Range:** 10 through 180 seconds

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

Example: Configuring RIPng Update Interval | 174

import (Protocols RIPng)

IN THIS SECTION

Syntax

```
import [ policy-names ];
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name protocols ripng group group-name neighbor neighbor-
name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng group group-name neighbor neighbor-name],
[edit protocols ripng],
[edit protocols ripng group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols ripng],
[edit routing-instances routing-instance-name protocols ripng group group-name neighbor neighbor-
name]
```

Description

Apply one or more policies to routes being imported into the local routing device from its neighbors.

Options

policy-names—Name of one or more policies.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

Example: Applying Policies to RIPng Routes Imported from Neighbors | 146

export

metric-in (Protocols RIPng)

IN THIS SECTION

Syntax

```
metric-in metric;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name protocols ripng group group-name neighbor neighbor-
name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng],
```

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng group group-name neighbor neighbor-name],
[edit protocols ripng],
[edit protocols ripng group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols ripng],
[edit routing-instances routing-instance-name protocols ripng group group-name neighbor neighbor-
name]
```

Description

Specify the metric to add to incoming routes when advertising into RIPng routes that were learned from other protocols. Use this statement to configure the routing device to prefer RIPng routes learned through a specific neighbor.

Options

metric—Metric value.

- **Range:** 1 through 16
- **Default:** 1

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

Example: Configuring the Metric Value Added to Imported RIPng Routes to Control the Route Selection Process | 167

metric-out (Protocols RIPng)

IN THIS SECTION

Syntax

```
metric-out metric;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng group group-name neighbor neighbor-
name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng group group-name neighbor neighbor-name],
[edit protocols ripng group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols ripng group group-name neighbor neighbor-
name]
```

Description

Specify the metric value to add to routes transmitted to the neighbor. Use this statement to control how other routing devices prefer RIPng routes sent from this neighbor.

When an export policy is configured, RIPng exports all learned routes to neighbors configured with the `neighbor` statement.

If a route being exported was learned from a member of the same RIPng group, the metric associated with that route (unless modified by an export policy) is the normal RIPng metric. For example, a RIPng route with a metric of 5 learned from a neighbor configured with a **metric-in** value of 2 is advertised with a combined metric of 7 when advertised to RIPng neighbors in the same group. However, if this route was learned from a RIPng neighbor in a different group or from a different protocol, the route is advertised with the metric value configured for that group with the `metric-out` statement. The default value for **metric-out** is 1.

To modify the metric for routes advertised outside a group, include the `metric-out` statement.

Options

metric—Metric value.

- **Range:** 1 through 16
- **Default:** 1

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

Example: Configuring the Metric Value Added to Imported RIPng Routes to Control the Route Selection Process | 167

Understanding RIPng Traffic Control with Metrics for Optimizing the Path Cost | 166

neighbor (Protocols RIPng)

IN THIS SECTION

Syntax

```
neighbor neighbor-name {
    import [ policy-names ];
    metric-in metric;
    receive <none>;
    route-timeout seconds;
    send <none>;
    update-interval seconds;
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng group group-name],
[edit protocols ripng group group-name],
[edit routing-instances routing-instance-name protocols ripng group group-name]
```

Description

Configure neighbor-specific RIPng parameters, thereby overriding the defaults set for the routing device.

Options

neighbor-name—Name of an interface over which a routing device communicates to its neighbors.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

preference (Protocols RIPng)

IN THIS SECTION

Syntax

```
preference preference;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng group group-name],
[edit protocols ripng group group-name],
[edit routing-instances routing-instance-name protocols ripng group group-name]
```

Description

Specify the preference of external routes learned by RIPng as compared to those learned from other routing protocols.

By default, Junos OS assigns a preference of 100 to routes that originate from RIPng. When Junos OS determines that a route is to become the active route, the software selects the route with the lowest preference and installs this route into the forwarding table.

To modify the default RIPng preference value, include the `preference` statement.

Options

preference—Preference value. A lower value indicates a more preferred route.

- **Range:** 0 through 4,294,967,295 (2^{32} – 1)
- **Default:** 100

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

receive (Protocols RIPng)

IN THIS SECTION

Syntax

```
receive <none>;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name protocols ripng group group-name neighbor neighbor-
name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng group group-name neighbor neighbor-name],
[edit protocols ripng],
[edit protocols ripng group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols ripng],
[edit routing-instances routing-instance-name protocols ripng group group-name neighbor neighbor-
name]
```

Description

Enable or disable receiving of update messages.

Options

none—(Optional) Disable receiving update messages.

- **Default:** Enabled

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

send (Protocols RIPng) | 305

ripng

IN THIS SECTION

Syntax

```
ripng {...}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols],
[edit protocols],
[edit routing-instances routing-instance-name protocols]
```

Description

Enable RIPng routing on the routing device.

Default

RIPng is disabled on the routing device.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

route-timeout (Protocols RIPng)

IN THIS SECTION

Syntax

```
route-timeout seconds;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng],
[edit protocols ripng],
[edit routing-instances routing-instance-name protocols ripng]
```

Description

Configure the route timeout interval for RIPng.

Options

seconds—Estimated time to wait before making updates to the routing table.

- **Range:** 30 through 360 seconds
- **Default:** 180 seconds

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 7.6.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

Example: Configuring RIPng Update Interval | 174

routing-instances (Multiple Routing Entities)

IN THIS SECTION

Syntax

```
routing-instances routing-instance-name { ... }
```

Hierarchy Level

```
[edit],
[edit logical-systems logical-system-name]
```

Description

Configure an additional routing entity for a router. You can create multiple instances of BGP, IS-IS, OSPF, OSPFv3, and RIP for a router. You can also create multiple routing instances for separating routing tables, routing policies, and interfaces for individual wholesale subscribers (retailers) in a Layer 3 wholesale network.

Each routing instance consist of the following:

- A set of routing tables
- A set of interfaces that belong to these routing tables
- A set of routing option configurations

Each routing instance has a unique name and a corresponding IP unicast table. For example, if you configure a routing instance with the name `my-instance`, its corresponding IP unicast table is my-instance.inet.0. All routes for `my-instance` are installed into my-instance.inet.0.

Routes are installed into the default routing instance inet.0 by default, unless a routing instance is specified.

In Junos OS Release 9.0 and later, you can no longer specify a routing-instance name of *primary*, *default*, or *bgp* or include special characters within the name of a routing instance.

In Junos OS Release 9.6 and later, you can include a slash (/) in a routing-instance name only if a logical system is not configured. That is, you cannot include the slash character in a routing-instance name if a logical system other than the default is explicitly configured. Routing-instance names, further, are restricted from having the form `__.*__` (beginning and ending with underscores). The colon `:` character cannot be used when multitopology routing (MTR) is enabled.

Default

Routing instances are disabled for the router.

Options

`routing-instance-name`	—Name of the routing instance. This must be a non-reserved string of not more than 128 characters.
`remote-vtep-list`	Configure static remote VXLAN tunnel endpoints.
`remote-vtep-v6-list`	Configure static IPv6 remote VXLAN tunnel endpoints.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

`remote-vtep-v6-list` statement introduced in Junos OS Release 17.3 for MX Series routers with MPC and MIC interfaces.

RELATED DOCUMENTATION

Example: Configuring Interprovider Layer 3 VPN Option A

Example: Configuring Interprovider Layer 3 VPN Option B

Example: Configuring Interprovider Layer 3 VPN Option C

send (Protocols RIPng)

IN THIS SECTION

Syntax

```
send <none>;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name protocols ripng group group-name neighbor neighbor-
name],
[edit logical-systems logical-system-name routing-instances routing-instances-name protocols
ripng],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng group group-name neighbor neighbor-name],
[edit protocols ripng],
[edit protocols ripng group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols ripng],
[edit routing-instances routing-instance-name protocols ripng group group-name neighbor neighbor-
name]
```

Description

Enable or disable sending of update messages.

Options

none—(Optional) Disable sending of update messages.

- **Default:** Enabled

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

receive (Protocols RIPng) | 297

traceoptions (Protocols RIPng)

IN THIS SECTION

Syntax

```
traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng],
[edit protocols ripng],
[edit routing-instances routing-instance-name protocols ripng]
```

Description

Set RIPng protocol-level tracing options.

Default

The default RIPng protocol-level trace options are inherited from the global `traceoptions` statement.

Options

disable—(Optional) Disable the tracing operation. One use of this option is to disable a single operation when you have defined a broad group of tracing operations, such as **all**.

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. We recommend that you place RIPng tracing output in the file **/var/log/ripng-log**.

files *number*—(Optional) Maximum number of trace files. When a trace file named ***trace-file*** reaches its maximum size, it is renamed ***trace-file*.0**, then ***trace-file*.1**, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum number of files, you must also specify a maximum file size with the **size** option.

- **Range:** 2 through 1000 files
- **Default:** 10 files

flag *flag*—Tracing operation to perform. To specify more than one tracing operation, include multiple `flag` statements.

RIPng Tracing Options

- **error**—RIPng error packets
- **expiration**—RIPng route expiration processing
- **holddown**—RIPng hold-down processing
- **nsr-synchronization**—Nonstop routing synchronization events
- **packets**—All RIPng packets
- **request**—RIPng information packets such as request, poll, and poll entry packets
- **trigger**—RIPng triggered updates
- **update**—RIPng update packets

Global Tracing Options

- **all**—All tracing operations
- **general**—A combination of the **normal** and **route** trace operations
- **normal**—All normal operations
- **Default:** If you do not specify this option, only unusual or abnormal operations are traced.
- **policy**—Policy operations and actions
- **route**—Routing table changes
- **state**—State transitions
- **task**—Routing protocol task processing
- **timer**—Routing protocol timer processing

flag-modifier—(Optional) Modifier for the tracing flag. You can specify one or more of these modifiers:

- **detail**—Provide detailed trace information.
- **receive**—Trace the packets being received.
- **receive-detail**—Provide detailed trace information for packets being received.
- **send**—Trace the packets being transmitted.
- **send-detail**—Provide detailed trace information for packets being transmitted.

no-world-readable—(Optional) Do not allow any user to read the log file.

size ***size***—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named ***trace-file*** reaches this size, it is renamed ***trace-file*.0**. When the ***trace-file*** again reaches its maximum size, ***trace-file*.0** is renamed ***trace-file*.1** and ***trace-file*** is renamed ***trace-file*.0**. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum file size, you must also specify a maximum number of trace files with the **files** option.

- **Syntax:** ***x*k** to specify KB, ***x*m** to specify MB, or ***x*g** to specify GB
- **Range:** 10 KB through the maximum file size supported on your system
- **Default:** 128 KB

world-readable—(Optional) Allow any user to read the log file.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

Example: Tracing RIPng Protocol Traffic | 185

update-interval (Protocols RIPng)

IN THIS SECTION

Syntax

```
update-interval seconds;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols ripng],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
ripng],
[edit protocols ripng],
[edit routing-instances routing-instance-name protocols ripng]
```

Description

Configure the interval at which routes learned by RIPng are sent to neighbors.

Options

seconds—Estimated time to wait before making updates to the routing table.

- **Range:** 10 through 60 seconds
- **Default:** 30 seconds

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

Release Information

Statement introduced in Junos OS Release 7.6.

Support for routing instances introduced in Junos OS Release 9.0.

RELATED DOCUMENTATION

Example: Configuring RIP Timers | 44

7

CHAPTER

RIP Operational Commands

clear rip general-statistics

IN THIS SECTION

Syntax

```
clear rip general-statistics
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches and QFX Series)

```
clear rip general-statistics
```

Description

Clear RIP general statistics.

Options

none	Clear RIP general statistics.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

clear

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear rip general-statistics

```
user@host> clear rip general-statistics
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show rip general-statistics | 342

clear rip statistics

IN THIS SECTION

Syntax

```
clear rip statistics
<instance (all | instance-name)>
<logical-system (all | logical-system-name)>
<neighbor>
<peer (all | address)>
```

Syntax (EX Series Switches and QFX Series)

```
clear rip statistics
<instance (all | instance-name)>
<neighbor>
```

Description

Clear RIP statistics.

Options

none	Reset RIP counters for all neighbors for all routing instances.
instance (all | *instance-name*)	(Optional) Clear RIP statistics for all instances or for the specified routing instance only.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
neighbor	(Optional) Clear RIP statistics for the specified neighbor only.
peer (all | *address*)	(Optional) Clear RIP statistics for a single peer or all peers.

Required Privilege Level

clear

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear rip statistics

```
user@host> clear rip statistics
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show rip statistics | 348

restart

IN THIS SECTION

Syntax

```
restart
<adaptive-services |ancpd-service | application-identification |audit-process | auto-
configuration |captive-portal-content-delivery |ce-l2tp-service | chassis-control  | class-of-
service |clksyncd-service |database-replication|datapath-trace-service |dhcp-service | diameter-
service | disk-monitoring | dynamic-flow-capture | ecc-error-logging | ethernet-connectivity-
fault-management |ethernet-link-fault-management |event-processing | firewall | general-
authentication-service | gracefully | iccp-service |idp-policy | immediately |interface-control
| ipsec-key-management | kernel-health-monitoring | kernel-replication | l2-learning | l2cpd-
service | l2tp-service | l2tp-universal-edge | lacp | license-service |link-management |local-
policy-decision-function |mac-validation |mib-process | mountd-service |mpls-traceroute |mspd |
multicast-snooping |named-service | nfsd-service | packet-triggered-subscribers |peer-selection-
service |pgm | pic-services-logging | pki-service |ppp | ppp-service | pppoe | protected-system-
domain-service | redundancy-interface-process | remote-operations | root-system-domain-service |
routing <logical-system logical-system-name>  | sampling | sbc-configuration-process | sdk-
service |service-deployment | services | snmp |soft |static-subscribers |statistics-service|
subscriber-management | subscriber-management-helper | tunnel-oamd |usb-control| vrrp |web-
management>
<gracefully | immediately | soft>
```

Syntax (ACX Series Routers)

```
restart
<adaptive-services |audit-process | auto-configuration | autoinstallation |chassis-control |
class-of-service |clksyncd-service |database-replication| dhcp-service | diameter-service | disk-
monitoring | dynamic-flow-capture | ethernet-connectivity-fault-management | ethernet-link-fault-
management |event-processing | firewall | general-authentication-service | gracefully |
immediately |interface-control | ipsec-key-management | l2-learning | lacp |link-management |mib-
process | mountd-service |mpls-traceroute |mspd | named-service | nfsd-service | pgm | pki-
service | ppp | pppoe | redundancy-interface-process | remote-operations | routing | sampling |
sdk-service |secure-neighbor-discovery | service-deployment | services | snmp |soft | statistics-
service| subscriber-management | subscriber-management-helper | tunnel-oamd | vrrp>
```

Syntax (EX Series Switches)

```
restart
<autoinstallation | chassis-control | class-of-service | database-replication | dhcp | dhcp-
service | diameter-service | dot1x-protocol | ethernet-link-fault-management | ethernet-
switching | event-processing | firewall | general-authentication-service | interface-control |
kernel-health-monitoring | kernel-replication | l2-learning | lacp | license-service | link-
management | lldpd-service | mib-process | mountd-service | multicast-snooping | pgm |
redundancy-interface-process | remote-operations | routing | secure-neighbor-discovery | service-
deployment | sflow-service | snmp | vrrp | web-management>
```

Syntax (MX Series Routers)

```
restart
<adaptive-services | ancpd-service | application-identification | audit-process | auto-
configuration | bbe-stats-service | captive-portal-content-delivery | ce-l2tp-service | chassis-
control | class-of-service | clksyncd-service | database-replication | datapath-trace-service |
dhcp-service | diameter-service | disk-monitoring | dynamic-flow-capture | ecc-error-logging |
ethernet-connectivity-fault-management | ethernet-link-fault-management | event-processing |
firewall | general-authentication-service | gracefully | iccp-service | idp-policy | immediately
|interface-control | ipsec-key-management |kernel-health-monitoring |  kernel-replication | l2-
learning | l2cpd-service | l2tp-service | l2tp-universal-edge | lacp | license-service | link-
management | local-policy-decision-function | mac-validation | mib-process | mountd-service |
mpls-traceroute | mspd | multicast-snooping |named-service | nfsd-service | packet-triggered-
subscribers |peer-selection-service | pgm | pic-services-logging | pki-service | ppp | ppp-
service | pppoe | protected-system-domain-service | redundancy-interface-process | remote-
operations | root-system-domain-service | routing | routing <logical-system logical-system-
name>  | sampling | sbc-configuration-process | sdk-service | service-deployment | services |
snmp |soft |static-subscribers |statistics-service| subscriber-management | subscriber-
management-helper | tunnel-oamd | usb-control | vrrp | web-management>
<all-members>
<gracefully | immediately | soft>
<local>
<member member-id>
```

Syntax (QFX Series)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dialer-services |
diameter-service | dlsw | ethernet-connectivity | event-processing | fibre-channel | firewall |
general-authentication-service | igmp-host-services | interface-control | ipsec-key-management |
isdn-signaling | l2ald | l2-learning | l2tp-service | mib-process | named-service | network-
access-service | nstrace-process | pgm | ppp | pppoe | redundancy-interface-process | remote-
operations |logical-system-name>  | routing | sampling |secure-neighbor-discovery | service-
deployment | snmp | usb-control | web-management>
<gracefully | immediately | soft>
```

Syntax (Routing Matrix)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | disk-monitoring |
dynamic-flow-capture | ecc-error-logging | event-processing | firewall | interface-control |
ipsec-key-management | kernel-replication | l2-learning | l2tp-service | lacp | link-management
| mib-process | pgm | pic-services-logging | ppp | pppoe | redundancy-interface-process | remote-
operations | routing <logical-system logical-system-name>  | sampling | service-deployment |
snmp>
<all | all-lcc | lcc number>
<gracefully | immediately | soft>
```

Syntax (SRX Series)

```
restart
<application-identification |application-security |audit-process |commitd-service |chassis-
control | class-of-service |database-replication |datapath-trace-service |ddns |dhcp |dhcp-
service |dynamic-flow-capture |disk-monitoring |event-processing | ethernet-connectivity-fault-
management |ethernet-link-fault-management |extensible-subscriber-services |fipsd |firewall |
firewall-authentication-service |general-authentication-service |gracefully |gprs-process |idp-
policy |immediately |interface-control | ipmi |ipsec-key-management |jflow-service |jnu-
management |jnx-wmicd-service |jsrp-service |kernel-replication |l2-learning |l2cpd-service |
lacp |license-service |logical-system-service |mib-process |mountd-service |named-service |
```

```
network-security |network-security-trace |nfsd-service |ntpd-service |pgm |pic-services-logging |
profilerd |pki-service |remote-operations |rest-api |routing |sampling |sampling-route-record |
scc-chassisd |secure-neighbor-discovery |security-intelligence |security-log |services |service-
deployment |simple-mail-client-service |soft |snmp |static-routed |statistics-service |
subscriber-management |subscriber-management-helper |system-log-vital |tunnel-oamd |uac-service |
user-ad-authentication |vrrp |web-management >
```

Syntax (TX Matrix Routers)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dhcp-service | diameter-
service | disk-monitoring | dynamic-flow-capture | ecc-error-logging | event-processing |
firewall | interface-control | ipsec-key-management | kernel-replication | l2-learning | l2tp-
service | lacp | link-management | mib-process |pgm | pic-services-logging | ppp | pppoe |
redundancy-interface-process | remote-operations | routing <logical-system logical-system-name>
| sampling | service-deployment | snmp| statistics-service>
<all-chassis | all-lcc | lcc number | scc>
<gracefully | immediately | soft>
```

Syntax (TX Matrix Plus Routers)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dhcp-service | diameter-
service | disk-monitoring | dynamic-flow-capture | ecc-error-logging | event-processing |
firewall | interface-control | ipsec-key-management | kernel-replication | l2-learning | l2tp-
service | lacp | link-management | mib-process | pgm | pic-services-logging | ppp | pppoe |
redundancy-interface-process | remote-operations | routing <logical-system logical-system-name>
| sampling | service-deployment | snmp| statistics-service>
<all-chassis | all-lcc | all-sfc | lcc number | sfc number>
<gracefully | immediately | soft>
```

Syntax (QFX Series)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dialer-services |
diameter-service | dlsw | ethernet-connectivity | event-processing | fibre-channel | firewall |
general-authentication-service | igmp-host-services | interface-control | ipsec-key-management |
isdn-signaling | l2ald | l2-learning | l2tp-service | mib-process | named-service | network-
access-service | nstrace-process | pgm | ppp | pppoe | redundancy-interface-process | remote-
operations |logical-system-name>  | routing | sampling |secure-neighbor-discovery | service-
deployment | snmp | usb-control | web-management>
<gracefully | immediately | soft>
```

Description

Restart a Junos OS process.

CAUTION: Never restart a software process unless instructed to do so by a customer support engineer. A restart might cause the router or switch to drop calls and interrupt transmission, resulting in possible loss of data.

The `restart` command expands all applications names including applications that are not required for the current platform. Therefore, a user could try to do a restart for an application that is not running for the current platform. This error message communicates that the restart failed because the application was not running on the system.

Options

none	Same as `gracefully`.
adaptive-services	(Optional) Restart the configuration management process that manages the configuration for stateful firewall, Network Address Translation (NAT), intrusion detection services (IDS), and IP Security (IPsec) services on the Adaptive Services PIC.

all-chassis	(TX Matrix and TX Matrix Plus routers only) (Optional) Restart the software process on all chassis.
all-lcc	(TX Matrix and TX Matrix Plus routers only) (Optional) For a TX Matrix router, restart the software process on all T640 routers connected to the TX Matrix router. For a TX Matrix Plus router, restart the software process on all T1600 routers connected to the TX Matrix Plus router.
all-members	(MX Series routers only) (Optional) Restart the software process for all members of the Virtual Chassis configuration.
all-sfc	(TX Matrix Plus routers only) (Optional) For a TX Matrix Plus router, restart the software processes for the TX Matrix Plus router (or switch-fabric chassis).
ancpd-service	(Optional) Restart the Access Node Control Protocol (ANCP) process, which works with a special Internet Group Management Protocol (IGMP) session to collect outgoing interface mapping events in a scalable manner.
application-identification	(Optional) Restart the process that identifies an application using intrusion detection and prevention (IDP) to allow or deny traffic based on applications running on standard or nonstandard ports.
application-security	(Optional) Restart the application security process.
audit-process	(Optional) Restart the RADIUS accounting process that gathers statistical data that can be used for general network monitoring, analyzing, and tracking usage patterns, for billing a user based on the amount of time or type of services accessed.
auto-configuration	(Optional) Restart the Interface Auto-Configuration process.
autoinstallation	(EX Series switches only) (Optional) Restart the autoinstallation process.
bbe-stats-service	(MX Series routers only) (Optional) Restart bbe-statsd, the BBE statistics collection and management process.
captive-portal-content-delivery	(Optional) Restart the HTTP redirect service by specifying the location to which a subscriber's initial Web browser session is redirected, enabling initial provisioning and service selection for the subscriber.
ce-l2tp-service	(M10, M10i, M7i, and MX Series routers only) (Optional) Restart the Universal Edge Layer 2 Tunneling Protocol (L2TP) process, which establishes L2TP tunnels and Point-to-Point Protocol (PPP) sessions through L2TP tunnels.
chassis-control	(Optional) Restart the chassis management process.

class-of-service	(Optional) Restart the class-of-service (CoS) process, which controls the router's or switch's CoS configuration.
clksyncd-service	(Optional) Restart the external clock synchronization process, which uses synchronous Ethernet (SyncE).
commitd-service	(Optional) Restart the committed services.
database-replication	(EX Series switches and MX Series routers only) (Optional) Restart the database replication process.
datapath-trace-service	(Optional) Restart the packet path tracing process.
dhcp	(EX Series switches only) (Optional) Restart the software process for a Dynamic Host Configuration Protocol (DHCP) server. A DHCP server allocates network IP addresses and delivers configuration settings to client hosts without user intervention.
dhcp-service	(Optional) Restart the Dynamic Host Configuration Protocol process.
dialer-services	(EX Series switches only) (Optional) Restart the ISDN dial-out process.
diameter-service	(Optional) Restart the diameter process.
disk-monitoring	(Optional) Restart disk monitoring, which checks the health of the hard disk drive on the Routing Engine.
dlsw	(QFX Series only) (Optional) Restart the data link switching (DLSw) service.
dot1x-protocol	(EX Series switches only) (Optional) Restart the port-based network access control process.
dynamic-flow-capture	(Optional) Restart the dynamic flow capture (DFC) process, which controls DFC configurations on Monitoring Services III PICs.
ecc-error-logging	(Optional) Restart the error checking and correction (ECC) process, which logs ECC parity errors in memory on the Routing Engine.
ethernet-connectivity-fault-management	(Optional) Restart the process that provides IEEE 802.1ag Operation, Administration, and Management (OAM) connectivity fault management (CFM) database information for CFM maintenance association end points (MEPs) in a CFM session.

ethernet-link-fault-management	(EX Series switches and MX Series routers only) (Optional) Restart the process that provides the OAM link fault management (LFM) information for Ethernet interfaces.
ethernet-switching	(EX Series switches only) (Optional) Restart the Ethernet switching process.
event-processing	(Optional) Restart the event process (eventd).
extensible-subscriber-services	(Optional) Restart the extensible subscriber services process.
fibre-channel	(QFX Series only) (Optional) Restart the Fibre Channel process.
fipsd	(Optional) Restart the fipsd services.
firewall	(Optional) Restart the firewall management process, which manages the firewall configuration and enables accepting or rejecting packets that are transiting an interface on a router or switch.
general-authentication-service	(EX Series switches and MX Series routers only) (Optional) Restart the general authentication process.
gprs-process	(Optional) Restart the General Packet Radio Service (GPRS) process.
gracefully	(Optional) Restart the software process.
iccp-service	(Optional) Restart the Inter-Chassis Communication Protocol (ICCP) process.
idp-policy	(Optional) Restart the intrusion detection and prevention (IDP) protocol process.
immediately	(Optional) Immediately restart the software process.
interface-control	(Optional) Restart the interface process, which controls the router's or switch's physical interface devices and logical interfaces.
ipmi	(Optional) Restart the intelligent platform management interface process.
ipsec-key-management	(Optional) Restart the IPsec key management process.
isdn-signaling	(QFX Series only) (Optional) Restart the ISDN signaling process, which initiates ISDN connections.
jflow-service	(Optional) Restart jflow service process.
jnu-management	(Optional) Restart jnu management process.

jnx-wmicd-service (Optional) Restart jnx wmicd service process.

jsrp-service (Optional) Restart the Juniper Services Redundancy Protocol (jsrdp) process, which controls chassis clustering.

kernel-health-monitoring (Optional) Restart the Routing Engine kernel health monitoring process, which enables health parameter data to be sent from kernel components to data collection applications. When you change the polling interval through `sysctl kern.jkhmd_polling_time_secs`, you must restart the kernel health monitoring process for the new polling interval to take effect.

kernel-replication (Optional) Restart the kernel replication process, which replicates the state of the backup Routing Engine when graceful Routing Engine switchover (GRES) is configured.

l2-learning (Optional) Restart the Layer 2 address flooding and learning process.

l2cpd-service (Optional) Restart the Layer 2 Control Protocol process, which enables features such as Layer 2 protocol tunneling and nonstop bridging.

l2tp-service (M10, M10i, M7i, and MX Series routers only) (Optional) Restart the Layer 2 Tunneling Protocol (L2TP) process, which sets up client services for establishing Point-to-Point Protocol (PPP) tunnels across a network and negotiating Multilink PPP if it is implemented.

l2tp-universal-edge (MX Series routers only) (Optional) Restart the L2TP process, which establishes L2TP tunnels and PPP sessions through L2TP tunnels.

lacp (Optional) Restart the Link Aggregation Control Protocol (LACP) process. LACP provides a standardized means for exchanging information between partner systems on a link to allow their link aggregation control instances to reach agreement on the identity of the LAG to which the link belongs, and then to move the link to that LAG, and to enable the transmission and reception processes for the link to function in an orderly manner.

lcc *number* (TX Matrix and TX Matrix Plus routers only) (Optional) For a TX Matrix router, restart the software process for a specific T640 router that is connected to the TX Matrix router. For a TX Matrix Plus router, restart the software process for a specific router that is connected to the TX Matrix Plus router.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.

- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

license-service (EX Series switches only) (Optional) Restart the feature license management process.

link-management (TX Matrix and TX Matrix Plus routers and EX Series switches only) (Optional) Restart the Link Management Protocol (LMP) process, which establishes and maintains LMP control channels.

lldpd-service (EX Series switches only) (Optional) Restart the Link Layer Discovery Protocol (LLDP) process.

local (MX Series routers only) (Optional) Restart the software process for the local Virtual Chassis member.

local-policy-decision-function (Optional) Restart the process for the Local Policy Decision Function, which regulates collection of statistics related to applications and application groups and tracking of information about dynamic subscribers and static interfaces.

logical-system-service (Optional) Restart the logical system service process.

mac-validation (Optional) Restart the Media Access Control (MAC) validation process, which configures MAC address validation for subscriber interfaces created on demux interfaces in dynamic profiles on MX Series routers.

member *member-id* (MX Series routers only) (Optional) Restart the software process for a specific member of the Virtual Chassis configuration. Replace `member-id` with a value of `0` or `1`.

mib-process (Optional) Restart the Management Information Base (MIB) version II process, which provides the router's MIB II agent.

mobile-ip (Optional) Restart the Mobile IP process, which configures Junos OS Mobile IP features.

mountd-service (EX Series switches and MX Series routers only) (Optional) Restart the service for NFS mount requests.

mpls-traceroute	(Optional) Restart the MPLS Periodic Traceroute process.
mspd	(Optional) Restart the Multiservice process.
multicast-snooping	(EX Series switches and MX Series routers only) (Optional) Restart the multicast snooping process, which makes Layer 2 devices, such as VLAN switches, aware of Layer 3 information, such as the media access control (MAC) addresses of members of a multicast group.
named-service	(Optional) Restart the DNS Server process, which is used by a router or a switch to resolve hostnames into addresses.
network-access-service	(QFX Series only) (Optional) Restart the network access process, which provides the router's Challenge Handshake Authentication Protocol (CHAP) authentication service.
network-security	(Optional) Restart the network security process.
network-security-trace	(Optional) Restart the network security trace process.
nfsd-service	(Optional) Restart the Remote NFS Server process, which provides remote file access for applications that need NFS-based transport.
ntpd-service	(Optional) Restart the Network Time Protocol (NTP) process.
packet-triggered-subscribers	(Optional) Restart the packet-triggered subscribers and policy control (PTSP) process, which allows the application of policies to dynamic subscribers that are controlled by a subscriber termination device.
peer-selection-service	(Optional) Restart the Peer Selection Service process.
pgcp-service	(Optional) Restart the pgcpd service process running on the Routing Engine. This option does not restart pgcpd processes running on mobile station PICs. To restart pgcpd processes running on mobile station PICs, use the `services pgcp gateway` option.
pgm	(Optional) Restart the process that implements the Pragmatic General Multicast (PGM) protocol for assisting in the reliable delivery of multicast packets.
pic-services-logging	(Optional) Restart the logging process for some PICs. With this process, also known as fsad (the file system access daemon), PICs send special logging information to the Routing Engine for archiving on the hard disk.
pki-service	(Optional) Restart the PKI Service process.

ppp	(Optional) Restart the Point-to-Point Protocol (PPP) process, which is the encapsulation protocol process for transporting IP traffic across point-to-point links.
ppp-service	(Optional) Restart the Universal edge PPP process, which is the encapsulation protocol process for transporting IP traffic across universal edge routers.
pppoe	(Optional) Restart the Point-to-Point Protocol over Ethernet (PPPoE) process, which combines PPP that typically runs over broadband connections with the Ethernet link-layer protocol that allows users to connect to a network of hosts over a bridge or access concentrator.
profilerd	(Optional) Restart the profiler process.
protected-system-domain-service	(Optional) Restart the Protected System Domain (PSD) process.
redundancy-interface-process	(Optional) Restart the ASP redundancy process.
remote-operations	(Optional) Restart the remote operations process, which provides the ping and traceroute MIBs.
rest-api	(Optional) Restart the rest api process.
root-system-domain-service	(Optional) Restart the Root System Domain (RSD) service.
routing	(ACX Series routers, QFX Series, EX Series switches, and MX Series routers only) (Optional) Restart the routing protocol process.
routing <logical-system *logical-system-name*>	(Optional) Restart the routing protocol process, which controls the routing protocols that run on the router or switch and maintains the routing tables. Optionally, restart the routing protocol process for the specified logical system only.
sampling	(Optional) Restart the sampling process, which performs packet sampling based on particular input interfaces and various fields in the packet header.
sampling-route-record	(Optional) Restart the sampling route record process.
sbc-configuration-process	(Optional) Restart the session border controller (SBC) process of the border signaling gateway (BSG).
scc	(TX Matrix routers only) (Optional) Restart the software process on the TX Matrix router (or switch-card chassis).

scc-chassisd	(Optional) Restart the scc chassisd process.
sdk-service	(Optional) Restart the SDK Service process, which runs on the Routing Engine and is responsible for communications between the SDK application and Junos OS. Although the SDK Service process is present on the router, it is turned off by default.
secure-neighbor-discovery	(QFX Series, EX Series switches, and MX Series routers only) (Optional) Restart the secure Neighbor Discovery Protocol (NDP) process, which provides support for protecting NDP messages.
security-intelligence	(Optional) Restart security intelligence process.
security-log	(Optional) Restart the security log process.
sfc *number*	(TX Matrix Plus routers only) (Optional) Restart the software process on the TX Matrix Plus router (or switch-fabric chassis). Replace `number` with `0`.
service-deployment	(Optional) Restart the service deployment process, which enables Junos OS to work with the Session and Resource Control (SRC) software.
services	(Optional) Restart a service.
services pgcp gateway *gateway-name*	(Optional) Restart the pgcpd process for a specific border gateway function (BGF) running on an MS-PIC. This option does not restart the pgcpd process running on the Routing Engine. To restart the pgcpd process on the Routing Engine, use the `pgcp-service` option.
sflow-service	(EX Series switches only) (Optional) Restart the flow sampling (sFlow technology) process.
simple-mail-client-service	(Optional) Restart the simple mail client service process.
snmp	(Optional) Restart the SNMP process, which enables the monitoring of network devices from a central location and provides the router's or switch's SNMP master agent.
soft	(Optional) Reread and reactivate the configuration without completely restarting the software processes. For example, BGP peers stay up and the routing table stays constant. Omitting this option results in a graceful restart of the software process.
static-routed	(Optional) Restart the static routed process.

static-subscribers	(Optional) Restart the static subscribers process, which associates subscribers with statically configured interfaces and provides dynamic service activation and activation for these subscribers.
statistics-service	(Optional) Restart the process that manages the Packet Forwarding Engine statistics.
subscriber-management	(Optional) Restart the Subscriber Management process.
subscriber-management-helper	(Optional) Restart the Subscriber Management Helper process.
system-log-vital	(Optional) Restart system log vital process.
tunnel-oamd	(Optional) Restart the Tunnel OAM process, which enables the Operations, Administration, and Maintenance of Layer 2 tunneled networks. Layer 2 protocol tunneling (L2PT) allows service providers to send Layer 2 protocol data units (PDUs) across the provider's cloud and deliver them to Juniper Networks EX Series Ethernet Switches that are not part of the local broadcast domain.
uac-service	(Optional) Restart the Unified Access Control (UAC) process.
usb-control	(MX Series routers) (Optional) Restart the USB control process.
user-ad-authentication	(Optional) Restart User ad Authentication process
vrrp	(ACX Series routers, EX Series switches, and MX Series routers only) (Optional) Restart the Virtual Router Redundancy Protocol (VRRP) process, which enables hosts on a LAN to make use of redundant routing platforms on that LAN without requiring more than the static configuration of a single default route on the hosts.
web-management	(QFX Series, EX Series switches, and MX Series routers only) (Optional) Restart the Web management process.

Required Privilege Level

reset

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

restart interfaces

```
user@host> restart interfaces
interfaces process terminated
interfaces process restarted
```

restart interface-control gracefully

```
user@host> restart interface-control gracefully
Interface control process started, pid 41129
```

restart interface-control (Junos OS Evolved)

```
user@host> restart interface-control
interface-control restart requested
Restarted aggd on re0
Restarted ifmand on re0
```

Release Information

Command introduced before Junos OS Release 7.4.

Options added:

- `dynamic-flow-capture` in Junos OS Release 7.4.
- `dlsw` in Junos OS Release 7.5.
- `event-processing` in Junos OS Release 7.5.

- `ppp` in Junos OS Release 7.5.
- `l2ald` in Junos OS Release 8.0.
- `link-management` in Junos Release 8.0.
- `pgcp-service` in Junos OS Release 8.4.
- `sbc-configuration-process` in Junos OS Release 9.5.
- `services pgcp gateway` in Junos OS Release 9.6.
- `sfc` and `all-sfc` for the TX Matrix Router in Junos OS Release 9.6.
- Command introduced before Junos OS Release 9.2 on SRX Series Firewalls.
- `bbe-stats-service` in Junos OS Release 18.4R1 on MX Series routers.
- `kernel-health-monitoring` in Junos OS Release 19.1R1.
- Introduced in Junos OS Evolved Release 19.1R1.

RELATED DOCUMENTATION

Overview of Operational Mode Commands

show policy

IN THIS SECTION

Syntax

```
show policy
<logical-system (all | logical-system-name)>
<policy-name>
<statistics >
```

Syntax (EX Series Switches)

```
show policy
<policy-name>
```

Description

Display information about configured routing policies.

Options

none	List the names of all configured routing policies.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
policy-name	(Optional) Show the contents of the specified policy.
statistics	(Optional) Use in conjunction with the `test policy` command to show the length of time (in microseconds) required to evaluate a given policy and the number of times it has been executed. This information can be used, for example, to help structure a policy so it is evaluated efficiently. Timers shown are per route; times are not cumulative. Statistics are incremented even when the router is learning (and thus evaluating) routes from peering routers.

Required Privilege Level

view

Output Fields

Table 8 on page 337 lists the output fields for the `show policy` command. Output fields are listed in the approximate order in which they appear.

Table 8: show policy Output Fields

Field Name	Field Description
policy-name	Name of the policy listed.
term	Name of the user-defined policy term. The term name `unnamed` is used for policy elements that occur outside of user defined terms
from	Match condition for the policy.
then	Action for the policy.

Sample Output

show policy

```
user@host> show policy
Configured policies:
__vrf-export-red-internal__
__vrf-import-red-internal__
red-export
rf-test-policy
multicast-scoping
```

show policy policy-name

```
user@host> show policy vrf-import-red-internal
Policy vrf-import-red-internal:
    from
         203.0.113.0/28  accept
         203.0.113.32/28  accept
    then reject
```

show policy statistics policy-name

```
user@host> show policy statistics iBGP-v4-RR-Import
Policy iBGP-v4-RR-Import:
    [1243328] Term Lab-Infra:
        from [1243328 0]  proto BGP
             [28 0] route filter:
                 10.11.0.0/8 orlonger
                 10.13.0.0/8 orlonger
        then [28 0] accept
    [1243300] Term External:
        from [1243300 1]  proto BGP
         [1243296 0]  community Ext-Com1 [64496:1515 ]
         [1243296 0]  prefix-list-filter Customer-Routes
         [1243296 0]  aspath AS6221
             [1243296 1] route filter:
                 172.16.49.0/12 orlonger
                 172.16.50.0/12 orlonger
                 172.16.51.0/12 orlonger
                 172.16.52.0/12 orlonger
                 172.16.56.0/12 orlonger
                 172.16.60.0/12 orlonger
        then [1243296 2] community + Ext-Com2 [64496:2000 ] [1243296 0] accept
    [4] Term Final:
        then [4 0] reject
```

Release Information

Command introduced before Junos OS Release 7.4.

`statistics` option introduced in Junos OS Release 16.1 for MX Series routers.

RELATED DOCUMENTATION

show policy damping

test policy

show policy conditions

IN THIS SECTION

Syntax

```
show policy conditions
<condition-name>
<detail>
<dynamic>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show policy conditions
<condition-name>
<detail>
<dynamic>
```

Description

Display all the configured conditions as well as the routing tables with which the configuration manager is interacting. If the `detail` keyword is included, the output also displays dependent routes for each condition.

Options

none	Display all configured conditions and associated routing tables.
condition-name	(Optional) Display information about the specified condition only.
detail	(Optional) Display the specified level of output.
dynamic	(Optional) Display information about the conditions in the dynamic database.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 9 on page 341 lists the output fields for the **show policy conditions** command. Output fields are listed in the approximate order in which they appear.

Table 9: show policy conditions Output Fields

Field Name	Field Description	Level of Output
Condition	Name of configured condition.	All levels
event	Condition type. If the `if-route-exists` option is configured, the event type is: Existence of a route in a specific routing table.	All levels
Dependent routes	List of routes dependent on the condition, along with the latest generation number.	`detail`
Condition tables	List of routing tables associated with the condition, along with the latest generation number and number of dependencies.	All levels
If-route-exists conditions	List of conditions configured to look for a route in the specified table.	All levels

Sample Output

show policy conditions detail

```
user@host> show policy conditions detail
Configured conditions:
Condition cond1, event: Existence of a route in a specific routing table
Dependent routes:
 172.16.4.4/32, generation 3
 6.6.6.6/32, generation 3
 10.10.10.10/32, generation 3

Condition cond2, event: Existence of a route in a specific routing table
```

```
Dependent routes:
None

Condition tables:
Table inet.0, generation 4, dependencies 3, If-route-exists conditions: cond1 (static) cond2
(static)
```

Release Information

Command introduced in Junos OS Release 9.0.

show rip general-statistics

IN THIS SECTION

Syntax

```
show rip general-statistics
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches and QFX Series)

```
show rip general-statistics
```

Description

Display brief RIP statistics.

Options

none	Display brief RIP statistics.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 10 on page 343 lists the output fields for the `show rip general-statistics` command. Output fields are listed in the approximate order in which they appear.

Table 10: show rip general-statistics Output Fields

Field Name	Field Description
bad msgs	Number of invalid messages received.

Table 10: show rip general-statistics Output Fields *(Continued)*

Field Name	Field Description
no recv intf	Number of packets received with no matching interface.
curr memory	Amount of memory currently used by RIP.
max memory	Most memory used by RIP.

Sample Output

show rip general-statistics

```
user@host> show rip general-statistics
RIPv2 I/O info:
     bad msgs       :         0
     no recv intf   :         0
     curr memory    :         0
     max memory     :         0
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

clear rip general-statistics | 315

show rip neighbor

IN THIS SECTION

Syntax

```
show rip neighbor
<instance (all | instance-name)>
<logical-system (all | logical-system-name)>
<name>
```

Syntax (EX Series Switches and QFX Series)

```
show rip neighbor
<instance (all | instance-name)>
<name>
```

Description

Display information about RIP neighbors.

Options

none	Display information about all RIP neighbors for all instances.
instance (all | *instance-name*)	(Optional) Display RIP neighbor information for all instances or for only the specified routing instance.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
name	(Optional) Display detailed information about only the specified RIP neighbor.

Required Privilege Level

view

Output Fields

Table 11 on page 347 lists the output fields for the `show rip neighbor` command. Output fields are listed in the approximate order in which they appear.

Table 11: show rip neighbor Output Fields

Field Name	Field Description
Neighbor	Name of the RIP neighbor. **NOTE**: Beginning with Junos OS Release 11.1, when you configure demand circuits, the output displays a demand circuit (DC) flag next to neighbor interfaces configured for demand circuits. If you configure demand circuits at the `[edit protocols rip group` *`group-name`* `neighbor` *`neighbor-name`*`]` hierarchy level, the output shows only the neighboring interface that you specifically configured as a demand circuit. If you configure demand circuits at the `[edit protocols rip group` *`group-name`*`]` hierarchy level, all of the interfaces in the group are configured as demand circuits. Therefore, the output shows all of the interfaces in that group as demand circuits.
State	State of the connection: **Up** or **Dn** (Down).
Source Address	Address of the port on the local router.
Destination Address	Address of the port on the remote router.
Send Mode	Send options: **broadcast**, **multicast**, **none**, or **version 1**.
Receive Mode	Type of packets to accept: **both**, **none**, **version 1**, or **version 2**.
In Met	Metric added to incoming routes when advertising into RIP routes that were learned from other protocols.

Sample Output

show rip neighbor

```
user@host> show rip neighbor
                  Local  Source          Destination     Send     Receive   In
Neighbor          State  Address         Address         Mode     Mode      Met
```

```
--------          -----  -------         -----------    ----    -------  ---
ge-2/3/0.0           Up  192.168.9.105   192.168.9.107  bcast   both       1
at-5/1/1.42          Dn  (null)          (null)         mcast   v2 only    3
at-5/1/0.42          Dn  (null)          (null)         mcast   both       3
at-5/1/0.0           Up  198.51.100.0    224.0.0.9      mcast   both       3
so-0/0/0.0           Up  192.168.9.97    224.0.0.9      mcast   both       3
```

show rip neighbor (With Demand Circuits Configured)

```
user@host> show rip neighbor
                  Local  Source          Destination    Send   Receive  In
Neighbor          State  Address         Address        Mode   Mode     Met
--------          -----  -------         -----------    ----   -------  ---
so-0/1/0.0(DC)       Up  10.10.10.2      224.0.0.9      mcast  both       1
so-0/2/0.0(DC)       Up  192.0.2.2       224.0.0.9      mcast  both       1
```

Release Information

Command introduced before Junos OS Release 7.4.

show rip statistics

IN THIS SECTION

Syntax

```
show rip statistics
<instance (all | instance-name)>
<logical-system (all | logical-system-name)>
<name>
<peer (all | address)>
```

Syntax (EX Series Switches and QFX Series)

```
show rip statistics
<instance (all | instance-name)>
<name>
```

Description

Display RIP statistics about messages sent and received on an interface, as well as information received from advertisements from other routing devices.

Options

none	Display RIP statistics for all routing instances.
instance (all | *instance-name*)	(Optional) Display RIP statistics for all instances or for only the specified routing instance.

logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
name	(Optional) Display detailed information about only the specified RIP neighbor.
peer (all | *address*)	(Optional) Display RIP statistics for a single peer or all peers.

Required Privilege Level

view

Output Fields

Table 12 on page 351 lists the output fields for the `show rip statistics` command. Output fields are listed in the approximate order in which they appear.

Table 12: show rip statistics Output Fields

Field Name	Field Description
RIP info	Information about RIP on the specified interface: • **port**—UDP port number used for RIP. • **update interval**—Interval between routing table updates, in seconds. • **holddown**—Hold-down interval, in seconds. • **timeout**—Timeout interval, in seconds. • **restart in progress**—Graceful restart status. Displayed when RIP is or has been in the process of graceful restart. • **restart time**—Estimated time for the graceful restart to finish, in seconds. • **restart will complete in**—Remaining time for the graceful restart to finish, in seconds. • **rts learned**—Number of routes learned through RIP. • **rts held down**—Number of routes held down by RIP. • **rqsts dropped**—Number of received request packets that were dropped. • **resps dropped**—Number of received response packets that were dropped.
logical-interface	Name of the logical interface and its statistics: • **routes learned**—Number of routes learned on the logical interface. • **routes advertised**—Number of routes advertised by the logical interface.

Table 12: show rip statistics Output Fields *(Continued)*

Field Name	Field Description
Counter	List of counter types: • **Updates Sent**—Number of update messages sent. • **Triggered Updates Sent**—Number of triggered update messages sent. • **Responses Sent**—Number of response messages sent. • **Bad Messages**—Number of invalid messages received. • **RIPv1 Updates Received**—Number of RIPv1 update messages received. • **RIPv1 Bad Route Entries**—Number of RIPv1 invalid route entry messages received. • **RIPv1 Updates Ignored**—Number of RIPv1 update messages ignored. • **RIPv2 Updates Received**—Number of RIPv2 update messages received. • **RIPv2 Bad Route Entries**—Number of RIPv2 invalid route entry messages received. • **RIPv2 Updates Ignored**—Number of RIPv2 update messages that were ignored. • **Authentication Failures**—Number of received update messages that failed authentication. • **RIP Requests Received**—Number of RIP request messages received. • **RIP Requests Ignored**—Number of RIP request messages ignored.
Total	Total number of packets for the selected counter.
Last 5 min	Number of packets for the selected counter in the most recent 5-minute period.
Last minute	Number of packets for the selected counter in the most recent 1-minute period.

Sample Output

show rip statistics

```
user@host> show rip statistics so-0/0/0.0
RIP info: port 520; update interval: 30s; holddown 180s; timeout 120s
restart in progress: restart time 60s; restart will complete in 55s
    rts learned  rts held down  rqsts dropped  resps dropped
              0              0              0              0
so-0/0/0.0:  0 routes learned; 501 routes advertised
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                        0            0            0
Triggered Updates Sent              0            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received              0            0            0
RIPv2 Bad Route Entries             0            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               0            0            0
RIP Requests Ignored                0            0            0
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

clear rip statistics | 317

show route

IN THIS SECTION

Syntax

```
show route
<all>
<destination-prefix>
<logical-system (all | logical-system-name)>
<private>
<te-ipv4-prefix-ip  te-ipv4-prefix-ip>
<te-ipv4-prefix-node-ip te-ipv4-prefix-node-ip>
<te-ipv4-prefix-node-iso te-ipv4-prefix-node-iso>
<te-ipv6-prefix-ipv6-addr te-ipv6-prefix-ipv6-addr>
<te-ipv6-prefix-node-iso te-ipv6-prefix-node-iso>
<rib-sharding (main | rib-shard-name)>
```

Syntax (EX Series Switches)

```
show route
<all>
<destination-prefix>
<private>
```

Description

Display the active entries in the routing tables.

Options

none	Display brief information about all active entries in the routing tables.
all	(Optional) Display information about all routing tables, including private, or internal, routing tables.
destination-prefix	(Optional) Display active entries for the specified address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
private	(Optional) Display information only about all private, or internal, routing tables.
programmed detail	(Optional) Display API-programmed routes.
display-client-data	(Optional) Display client id and cookie information for routes installed by the routing protocol process client applications.
te-ipv4-prefix-ip *te-ipv4-prefix-ip*	(Optional) Display IPv4 address of the traffic-engineering prefix, without the mask length if present in the routing table.

te-ipv4-prefix-node-ip *te-ipv4-prefix-node-ip*	(Optional) Display all prefixes that have originated from the traffic-engineering node. You can filter IPv4 node addresses from the traffic-engineered routes in the `lsdist.0` table.
te-ipv4-prefix-node-iso *te-ipv4-prefix-node-iso*	(Optional) Display all prefixes that have originated from the traffic-engineering node. You can filter IPv4 routes with the specified ISO circuit ID from the `lsdist.0` table.
te-ipv6-prefix-ipv6-addr *te-ipv6-prefix-ipv6-addr*	(Optional) Filter IPv6 node addresses from the traffic-engineering IPv6 prefix.
te-ipv6-prefix-node-iso *te-ipv6-prefix-node-iso*	(Optional) Filter IPv6 routes with the specified ISO circuit ID in the traffic-engineering IPv6 prefix.
rib-sharding (main | *rib-shard-name*)	(Optional) Display the rib shard name.

Required Privilege Level

view

Output Fields

Table 13 on page 356 describes the output fields for the `show route` command. Output fields are listed in the approximate order in which they appear.

Table 13: show route Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).
number destinations	Number of destinations for which there are routes in the routing table.

Table 13: show route Output Fields *(Continued)*

Field Name	Field Description
number routes	Number of routes in the routing table and total number of routes in the following states: • `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. 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. If you have configured `uRPF-loose` mode, the holddown bit is most likely set because Kernel Routing Table (KRT) is using inactive route to build valid incoming interfaces. In this case, you can ignore the holddown state because nothing is wrong. • `hidden` (routes that are not used because of a routing policy).

Table 13: show route Output Fields *(Continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th></tr>
<tr><td>destination-prefix</td><td>Route destination (for example:10.0.0.1/24). Sometimes the route information is presented in another format, such as:
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):
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.</td></tr>
<tr><td>[protocol, preference]</td><td>Protocol from which the route was learned and the preference value for the route.
<code>+</code>—A plus sign indicates the active route, which is the route installed from the routing table into the forwarding table.
<code>-</code> —A hyphen indicates the last active route.
<code>*</code>—An asterisk indicates that the route is both the active and the last active route. An asterisk before a <code>to</code> line indicates the best subpath to the route.
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.</td></tr>
</table>

Table 13: show route Output Fields *(Continued)*

Field Name	Field Description
weeks:days hours:minutes:seconds	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	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: • `I`—IGP. • `E`—EGP. • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—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. **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.

Table 13: show route Output Fields *(Continued)*

Field Name	Field Description
encapsulated	Extended next-hop encoding capability enabled for the specified BGP community for routing IPv4 traffic over IPv6 tunnels. When BGP receives routes without the tunnel community, IPv4-0ver IPv6 tunnels are not created and BGP routes are resolved without encapsulation.
Route Labels	Stack of labels carried in the BGP route update.
validation-state	(BGP-learned routes) Validation status of the route: • 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. • Unverified—Indicates that the origin of the prefix is not verified against the database. This is because the database got populated and the validation is not called for in the BGP import policy, although origin validation is enabled, or the origin validation is not enabled for the BGP peers. • Valid—Indicates that the prefix and autonomous system pair are found in the database.
to	Next hop to the destination. An angle bracket (>) indicates that the route is the selected route. If the destination is `Discard`, traffic is dropped.

Table 13: show route Output Fields *(Continued)*

Field Name	Field Description
via	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: • 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). For VPNs, expect to see multiple push operations, corresponding to the inner and outer labels required for VPN routes (in the case of a direct PE-to-PE connection, the VPN route would have the inner label push only).
Private unicast	(Enhanced subscriber management for MX Series routers) Indicates that an access-internal route is managed by enhanced subscriber management. By contrast, access-internal routes not managed by enhanced subscriber management are displayed with associated next-hop and media access control (MAC) address information.
balance	Distribution of the load based on the underlying operational interface bandwidth for equal-cost multipaths (ECMP) across the nexthop gateways in percentages.

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

The following sample output shows route hierarchy for translation route.

```
user@host> show route 10.1.1.1

C1.inet.0: 44 destinations, 44 routes (44 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.1.1.1/32        *[PRPD/10] 00:16:50, metric 2
                    >  to 192.0.2.2 via ge-0/0/1.0
```

show route forwarding-table matching 10.1.1.1

```
user@host> show route forwarding-table matching 10.1.1.1
Routing table: C1.inet
Internet:
Destination        Type RtRef Next hop           Type Index    NhRef Netif

10.1.1.1/32        user     0                    indr  1048574     4

                                                 comp      624     2
```

show route 10.1.1.1 extensive expanded-nh

```
user@host> show route 10.1.1.1extensive expanded-nh
C1.inet
C1.inet.0: 44 destinations, 44 routes (44 active, 0 holddown, 0 hidden)
10.1.1.1/32 (1 entry, 1 announced)
Installed-nexthop:
Indr (0xc5c207c) ::44.0.0.1
  Krt_inh (0xc6fd004) Index:1048574 PNH: ::44.0.0.1
    Translate-comp (0xc5c2144) Index:624 v4tov6 src ::22.0.0.1 dest ::44.0.0.1
```

show route te-ipv6-prefix-ipv6-addr

```
user@host> show route te-ipv6-prefix-ipv6-addr 10::10

inet.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)

inet.3: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

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

mpls.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)

inet6.0: 10 destinations, 11 routes (10 active, 0 holddown, 0 hidden)

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

lsdist.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
```

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

IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:10::10/128 } ISIS-L1:0 }/1216
                   *[IS-IS/15] 00:07:58
                       Fictitious
```

show route te-ipv6-prefix-node-iso

```
user@host> show route te-ipv6-prefix-node-iso 0100.0100.0100.00

inet.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)

inet.3: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

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

mpls.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)

inet6.0: 10 destinations, 11 routes (10 active, 0 holddown, 0 hidden)

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

lsdist.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:10::10/128 } ISIS-L1:0 }/
1216
                   *[IS-IS/15] 00:08:46
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:21:0:1::1/128 } ISIS-L1:0 }/
1216
                   *[IS-IS/15] 00:08:46
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:abcd::128:207:200:16/128 } ISIS-
L1:0 }/1216
                   *[IS-IS/15] 00:08:46
                       Fictitious
```

show route (VPN)

The following sample output shows a VPN route with composite next hops enabled. The first `Push` operation corresponds to the outer label. The second `Push` operation corresponds to the inner label.

```
user@host> show route 192.0.2.0

13979:665001.inet.0: 871 destinations, 3556 routes (871 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.0.2.0/24        [BGP/170] 00:28:32, localpref 100, from 10.9.9.160
                      AS path: 13980 ?, validation-state: unverified
                    > to 10.100.0.42 via ae2.0, Push 16, Push 300368(top)
                    [BGP/170] 00:28:28, localpref 100, from 10.9.9.169
                      AS path: 13980 ?, validation-state: unverified
                    > to 10.100.0.42 via ae2.0, Push 126016, Push 300368(top)
                   #[Multipath/255] 00:28:28, metric2 102
                    > to 10.100.0.42 via ae2.0, Push 16, Push 300368(top)
                      to 10.100.0.42 via ae2.0, Push 16, Push 300368(top)
```

show route (with Destination Prefix)

```
user@host> show route 192.168.0.0/12

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

192.168.0.0/12      *[Static/5] 2w4d 12:54:27
                    > to 192.168.167.254 via fxp0.0
```

show route destination-prefix detail

```
user@host> show route 198.51.100.0 detail

inet.0: 15 destinations, 20 routes (15 active, 0 holddown, 0 hidden)
198.51.100.0/24 (2 entries, 2 announced)
        *BGP    Preference: 170/-101
         ...
         BGP-Static Preference: 4294967292
```

```
          Next hop type: Discard
          Address: 0x9041ae4
          Next-hop reference count: 2
          State: <NoReadvrt Int Ext AlwaysFlash>
          Inactive reason: Route Preference
          Local AS:   200
          Age: 4d 1:40:40
          Validation State: unverified
          Task: RT
          Announcement bits (1): 2-BGP_RT_Background
          AS path: 4 5 6 I
```

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 203.0.113.1
           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: 64510 Peer AS: 64511
           Age: 3  Metric2: 1
           Validation State: unverified
           Task: BGP_64510.10.0.0.30+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:64502:100 encapsulation:0L:14
           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 programmed detail

```
user@host> show route programmed detail
inet.0: 36 destinations, 37 routes (36 active, 0 holddown, 0 hidden)
100.75.1.0/27 (2 entries, 1 announced)
        *Static Preference: 5/100
                Next hop type: Router, Next hop index: 0
                Address: 0xcc38a10
                Next-hop reference count: 1
                Next hop: 100.30.1.2 via ge-0/0/2.0 weight 0x1, selected
                Session Id: 0x0
                Next hop: via fti0.1001 weight 0x8001
                Session Id: 0x0
               State: <Active Int NSR-incapable Programmed>
                Age: 37
                Validation State: unverified
                Announcement bits (1): 0-KRT
                AS path: I
```

Release Information

Command introduced before Junos OS Release 7.4.

Option `private` introduced in Junos OS Release 9.5.

Option `private` introduced in Junos OS Release 9.5 for EX Series switches.

Option `display-client-data` introduced in Junos OS Release 16.2R1 on MX80, MX104, MX240, MX480, MX960, MX2010, MX2020, vMX Series routers.

Options `te-ipv4-prefix-ip`, `te-ipv4-prefix-node-ip`, and `te-ipv4-prefix-node-iso` introduced in Junos OS Release 17.2R1 on MX Series and PTX Series.

`rib-sharding` option introduced in cRPD Release 20.1R1.

RELATED DOCUMENTATION

Understanding IS-IS Configuration
Verifying and Managing Junos OS Enhanced Subscriber Management

show route active-path

IN THIS SECTION

Syntax

```
show route active-path
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route active-path
<brief | detail | extensive | terse>
```

Description

Display all active routes for destinations. An active route is a route that is selected as the best path. Inactive routes are not displayed.

Options

none	Display all active routes.
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 | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 active-path

```
user@host> show route active-path

inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.70.19/32    *[Direct/0] 21:33:52
                    > via lo0.0
10.255.71.50/32    *[IS-IS/15] 00:18:13, metric 10
                    > to 172.16.100.1 via so-2/1/3.0
172.16.100.1/24       *[Direct/0] 00:18:36
                    > via so-2/1/3.0
172.16.100.1/32       *[Local/0] 00:18:41
                      Local via so-2/1/3.0
192.168.64.0/21    *[Direct/0] 21:33:52
                    > via fxp0.0
192.168.70.19/32   *[Local/0] 21:33:52
                      Local via fxp0.0
```

show route active-path brief

The output for the `show route active-path brief` command is identical to that for the `show route active-path` command. For sample output, see `show route active-path`.

show route active-path detail

```
user@host> show route active-path detail

inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)

10.255.70.19/32 (1 entry, 1 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via lo0.0, selected
                State: <Active Int>
                Local AS:   200
```

```
                Age: 21:37:10
                Task: IF
                Announcement bits (3): 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

10.255.71.50/32 (1 entry, 1 announced)
        *IS-IS  Preference: 15
                Level: 1
                Next hop type: Router, Next hop index: 397
                Next-hop reference count: 4
                Next hop: 172.16.100.1 via so-2/1/3.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:31      Metric: 10
                Task: IS-IS
                Announcement bits (4): 0-KRT 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

172.16.100.0/24 (1 entry, 1 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via so-2/1/3.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:54
                Task: IF
                Announcement bits (3): 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

172.16.100.1/32 (1 entry, 1 announced)
        *Local  Preference: 0
                Next hop type: Local
                Next-hop reference count: 11
                Interface: so-2/1/3.0
                State: <Active NoReadvrt Int>
                Local AS:   200
                Age: 21:59
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

192.168.64.0/21 (1 entry, 1 announced)
```

```
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via fxp0.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:37:10
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

192.168.70.19/32 (1 entry, 1 announced)
        *Local  Preference: 0
                Next hop type: Local
                Next-hop reference count: 11
                Interface: fxp0.0
                State: <Active NoReadvrt Int>
                Local AS:   200
                Age: 21:37:10
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I
```

show route active-path extensive

```
user@host> show route active-path extensive

inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)
10.255.70.19/32 (1 entry, 1 announced)
TSI:
IS-IS level 1, LSP fragment 0
IS-IS level 2, LSP fragment 0
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via lo0.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:39:47
                Task: IF
                Announcement bits (3): 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
```

```
                AS path: I

10.255.71.50/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 10.255.71.50/32 -> {172.16.100.1}
IS-IS level 2, LSP fragment 0
        *IS-IS  Preference: 15
                Level: 1
                Next hop type: Router, Next hop index: 397
                Next-hop reference count: 4
                Next hop: 172.16.100.1 via so-2/1/3.0, selected
                State: ‹Active Int›
                Local AS:   200
                Age: 24:08      Metric: 10
                Task: IS-IS
                Announcement bits (4): 0-KRT 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

172.16.100.1/24 (1 entry, 1 announced)
TSI:
IS-IS level 1, LSP fragment 0
IS-IS level 2, LSP fragment 0
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via so-2/1/3.0, selected
                State: ‹Active Int›
                Local AS:   200
                Age: 24:31
                Task: IF
                Announcement bits (3): 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

172.16.100.1/32 (1 entry, 1 announced)
        *Local  Preference: 0
                Next hop type: Local
                Next-hop reference count: 11
                Interface: so-2/1/3.0
                State: ‹Active NoReadvrt Int›
                Local AS:   200
                Age: 24:36
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
```

```
                AS path: I

192.168.64.0/21 (1 entry, 1 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via fxp0.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:39:47
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

192.168.70.19/32 (1 entry, 1 announced)
        *Local  Preference: 0
                Next hop type: Local
                Next-hop reference count: 11
                Interface: fxp0.0
                State: <Active NoReadvrt Int>
                Local AS:   200
                Age: 21:39:47
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I
```

show route active-path terse

```
user@host> show route active-path terse

inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 10.255.70.19/32    D   0                       >lo0.0
* 10.255.71.50/32    I  15         10            >172.16.100.1.
* 172.16.100.0/24       D   0                       >so-2/1/3.0
* 172.16.100.2/32       L   0                        Local
* 192.168.64.0/21    D   0                       >fxp0.0
```

```
* 192.168.70.19/32   L    0                        Local
```

Release Information

Command introduced in Junos OS Release 8.0.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route advertising-protocol

IN THIS SECTION

Syntax

```
show route advertising-protocol protocol neighbor-address
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Description

Display the routing information as it has been prepared for advertisement to a particular neighbor of a particular dynamic routing protocol.

Options

brief | detail | extensive | terse	(Optional) Display the specified level of output.
logical-system (all | ***logical-system-name*****)**	(Optional) Perform this operation on all logical systems or on a particular logical system.
neighbor-address	Address of the neighboring router to which the route entry is being transmitted.
protocol	Protocol transmitting the route: • `bgp`—Border Gateway Protocol • `dvmrp`—Distance Vector Multicast Routing Protocol • `msdp`—Multicast Source Discovery Protocol • `pim`—Protocol Independent Multicast • `rip`—Routing Information Protocol • `ripng`—Routing Information Protocol next generation

Additional Information

Routes displayed are routes that the routing table has exported into the routing protocol and that have been filtered by the associated protocol's `export` routing policy statements. Starting with Junos OS Release 13.3, you can display the routing instance table `foo` for any address family, on a VPN route reflector, or a VPN AS boundary router that is advertising local VPN routes. However, If you do not specify the `table` in the command, the output displays each VRF prefix twice.

Required Privilege Level

view

Output Fields

Table 14 on page 377 lists the output fields for the `show route advertising-protocol` command. Output fields are listed in the approximate order in which they appear.

Table 14: show route advertising-protocol Output Fields

Field Name	Field Description	Level of Output
routing-table-name	Name of the routing table—for example, inet.0.	All levels
number destinations	Number of destinations for which there are routes in the routing table.	All levels
number routes	Number of routes in the routing table and total number of routes in the following states: • `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)	All levels

Table 14: show route advertising-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Prefix	Destination prefix.	`brief` none
destination-prefix (entry , announced)	Destination prefix. 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.	`detail` `extensive`
BGP group and type	BGP group name and type (Internal or External).	`detail` `extensive`
Route Distinguisher	Unique 64-bit prefix augmenting each IP subnet.	`detail` `extensive`
Advertised Label	Incoming label advertised by the Label Distribution Protocol (LDP). When an IP packet enters a label-switched path (LSP), the ingress router examines the packet and assigns it a label based on its destination, placing the label in the packet's header. The label transforms the packet from one that is forwarded based on its IP routing information to one that is forwarded based on information associated with the label.	`detail` `extensive`
Label-Base, range	First label in a block of labels and label block size. A remote PE router uses this first label when sending traffic toward the advertising PE router.	`detail` `extensive`
VPN Label	Virtual private network (VPN) label. Packets are sent between CE and PE routers by advertising VPN labels. VPN labels transit over either a Resource Reservation Protocol (RSVP) or a Label Distribution Protocol (LDP) label-switched path (LSP) tunnel.	`detail` `extensive`
Nexthop	Next hop to the destination. An angle bracket (>) indicates that the route is the selected route. If the next-hop advertisement to the peer is Self, and the RIB-out next hop is a specific IP address, the RIB-out IP address is included in the extensive output.	All levels

Table 14: show route advertising-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
MED	Multiple exit discriminator value included in the route.	`brief`
Lclpref or Localpref	Local preference value included in the route.	All levels
Queued	When BGP route prioritization is enabled and a route is present in a priority queue, this shows which priority queue the route is in.	All levels except `brief`
AS path	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: • `I`—IGP. • `E`—EGP. • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—Brackets enclose the local AS number associated with the AS path if configured on the router, 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. **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.	All levels
Route Labels	Stack of labels carried in the BGP route update.	`detail` `extensive`

Table 14: show route advertising-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Cluster list	(For route reflected output only) Cluster ID sent by the route reflector.	`detail` `extensive`
Originator ID	(For route reflected output only) Address of routing device that originally sent the route to the route reflector.	`detail` `extensive`
`Communities`	Community path attribute for the route. See the output field table for the *show route detail* command for all possible values for this field.	`detail` `extensive`
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.	`detail` `extensive`
Attrset AS	Number, local preference, and path of the autonomous system (AS) that originated the route. These values are stored in the Attrset attribute at the originating router.	`detail` `extensive`
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).	`detail` `extensive`
control flags	Control flags: `none` or `Site Down`.	`detail` `extensive`
mtu	Maximum transmission unit (MTU) of the Layer 2 circuit.	`detail` `extensive`

Sample Output

show route advertising-protocol bgp (Layer 3 VPN)

```
user@host> show route advertising-protocol bgp 10.255.14.171
      VPN-A.inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
      Prefix             Nexthop                MED    Lclpref AS path
      10.255.14.172/32   Self                     1        100 I
      VPN-B.inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
```

```
    Prefix             Nexthop                 MED     Lclpref AS path
    10.255.14.181/32   Self                      2        100 I
```

show route advertising-protocol bgp (IPv6 unicast address community)

```
user@host> show route advertising-protocol bgp 10.255.165.220 extensive
       vpn1.inet.0: 13 destinations, 13 routes (13 active, 0 holddown, 0 hidden)
    * 128.205.172.129/32 (1 entry, 1 announced)
    BGP group internal type Internal
    Route Distinguisher: 10.255.168.42:8
    VPN Label: 299808
    Nexthop: Self
    Flags: Nexthop Change
    MED: 1
    Localpref: 100
       AS path: [203] I
       Communities: rte-type:0.0.0.0:1:0 ipv6-origin:<2001:db8:9999::9>:89
       ipv6-target:<2001:db8:9999::9>:89 ipv6-extended:203:<2001:db8:9999::9>:89
       ipv6-origin:<2001:db8:9999::9>:137 ipv6-target:<2001:db8:9999::9>:137
       ipv6-extended:515:<2001:db8:9999::9>:137
```

show route advertising-protocol bgp detail

```
user@host> show route advertising-protocol bgp 10.222.1.3 detail
bgp20.inet.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
10.222.1.11/32 (1 entry, 1 announced)
 BGP group pe-pe type Internal
     Route Distinguisher: 10.255.14.11:69
     Advertised Label: 100000
     next hop: Self
     Localpref: 100
     AS path: 2 I
     Communities: target:69:20
     AIGP 210
10.8.0.0/16 (1 entry, 1 announced)
 BGP group pe-pe type Internal
     Route Distinguisher: 10.255.14.11:69
     Advertised Label: 100000
     Next hop: Self
     Localpref: 100
```

```
     AS path: 2 I
     Communities: target:69:20
     AIGP 210
```

show route advertising-protocol bgp detail (Aggregate Extended Community Bandwidth)

```
user@host> show route advertising-protocol bgp 10.0.4.2 10.0.2.0/30 detail
inet.0: 20 destinations, 26 routes (20 active, 0 holddown, 0 hidden)
* 10.0.2.0/30 (2 entries, 1 announced)
 BGP group external2 type External
     Nexthop: Self
     AS path: [65000] 65001 I
     Communities: bandwidth:65000:80000000
```

show route advertising-protocol bgp detail (BGP Multicast)

```
user@host>show route advertising-protocol bgp 10.4.6.1 detail
bgpmcast.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
* 4:3:0:0:32:10.2.20.20:32:225.1.1.1:10.1.1.1:10.20.20.20/240 (1 entry, 1 announced)
 BGP group ibgp type Internal
     Nexthop: 10.4.6.6
     Localpref: 100
     AS path: [65000] 65001 I
     Communities: target:65000:100
     Tunnel type: AnyEncap, RPF tunnel:, Remote end point: 10.1.2.1
     Tunnel type: AnyEncap, Remote end point: 10.1.4.1
```

show route advertising-protocol bgp detail (Labeled Unicast)

```
user@host>show route advertising bgp 10.1.1.3 detail
inet.0: 69 destinations, 70 routes (69 active, 0 holddown, 0 hidden)
* 10.1.1.8/32 (2 entries, 2 announced)
BGP group ibgp type Internal
Route Labels: 1000123(top) 1000124 1000125 1000126
Nexthop: 10.1.1.4
MED: 7
Localpref: 100
AS path: [65005] I
```

```
Cluster ID: 10.3.3.3
Originator ID: 10.1.1.1
Entropy label capable
inet6.0: 26 destinations, 28 routes (26 active, 0 holddown, 0 hidden)
* 2001:db8:100::1/128 (2 entries, 1 announced)
BGP group ibgp type Internal
Labels: 1000123(top) 1000124 1000125 1000126
Nexthop: 2001:db8:0:ffff:10.1.1.4
Localpref: 100
AS path: [65005] I
Cluster ID: 10.3.3.3
Originator ID: 10.1.1.1
```

show route advertising-protocol bgp detail (Layer 2 VPN)

```
user@host> show route advertising-protocol bgp 192.168.24.1 detail
vpn-a.l2vpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
192.168.16.10:1:1:1/96 (1 entry, 1 announced)
 BGP group int type Internal
     Route Distinguisher: 192.168.16.1:1
     Label-base : 32768, range : 3
     Nexthop: Self
     Localpref: 100
     AS path: I
     Communities: target:65412:100
     AIGP 210
     Layer2-info: encaps:VLAN, control flags:, mtu:
```

show route advertising-protocol bgp detail (Layer 3 VPN)

```
user@host> show route advertising-protocol bgp 10.255.14.176 detail
vpna.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
* 10.49.0.0/30 (1 entry, 1 announced)
 BGP group ibgp type Internal
     Route Distinguisher: 10.255.14.174:2
     VPN Label: 101264
     Nexthop: Self
     Localpref: 100
     AS path: I
     Communities: target:10:100
```

```
      AIGP 210
      AttrSet AS: 65100
          Localpref: 100
          AS path: I
...
```

show route advertising-protocol bgp extensive all (Next Hop Self with RIB-out IP Address)

```
user@host> show route advertising-protocol bgp 10.200.0.2 10.170.1.0/24 extensive all
inet.0: 13 destinations, 19 routes (13 active, 0 holddown, 6 hidden)
  10.170.1.0/24 (2 entries, 1 announced)
 BGP group eBGP-INTEROP type External
      Nexthop: Self (rib-out 10.100.3.2)
      AS path: [64713] 65200 I
...
```

Release Information

Command introduced before Junos OS Release 7.4.

`ipv6-origin`, `ipv6-target`, and `ipv6-extended Community` output fields supported in Junos OS Release and Junos OS Evolved Release 23.1.

RELATED DOCUMENTATION

Example: Configuring the MED Attribute That Determines the Exit Point in an AS

show route all

IN THIS SECTION

Syntax

```
show route all
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route all
```

Description

Display information about all routes in all routing tables, including private, or internal, tables.

Options

none	Display information about all routes in all routing tables, including private, or internal, tables.

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

Required Privilege Level

view

Output Fields

In Junos OS Release 9.5 and later, only the output fields for the `show route all` command display all routing tables, including private, or hidden, routing tables. The output field table of the *show route* command does not display entries for private, or hidden, routing tables in Junos OS Release 9.5 and later.

Sample Output

show route all

The following example displays a snippet of output from the `show route` command and then displays the same snippet of output from the `show route all` command:

```
user@host> show route
mpls.0: 7 destinations, 7 routes (5 active, 0 holddown, 2 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
0                  *[MPLS/0] 2d 02:24:39, metric 1
                      Receive
1                  *[MPLS/0] 2d 02:24:39, metric 1
                      Receive
2                  *[MPLS/0] 2d 02:24:39, metric 1
                      Receive
800017             *[VPLS/7] 1d 14:00:16
                    > via vt-3/2/0.32769, Pop
800018             *[VPLS/7] 1d 14:00:26
                    > via vt-3/2/0.32772, Pop
```

```
user@host> show route all
mpls.0: 7 destinations, 7 routes (5 active, 0 holddown, 2 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
0                  *[MPLS/0] 2d 02:19:12, metric 1
                      Receive
1                  *[MPLS/0] 2d 02:19:12, metric 1
                      Receive
2                  *[MPLS/0] 2d 02:19:12, metric 1
                      Receive
800017             *[VPLS/7] 1d 13:54:49
                    > via vt-3/2/0.32769, Pop
800018             *[VPLS/7] 1d 13:54:59
                    > via vt-3/2/0.32772, Pop
vt-3/2/0.32769      [VPLS/7] 1d 13:54:49
                      Unusable
vt-3/2/0.32772      [VPLS/7] 1d 13:54:59
                      Unusable
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route brief

show route detail

show route best

IN THIS SECTION

Syntax

```
show route best destination-prefix
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route best destination-prefix
<brief | detail | extensive | terse>
```

Description

Display the route in the routing table that is the best route to the specified address or range of addresses. The best route is the longest matching route.

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`.
destination-prefix	Address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 best

```
user@host> show route best 10.255.70.103
inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
10.255.70.103/32   *[OSPF/10] 1d 13:19:20, metric 2
                    > to 10.31.1.6 via ge-3/1/0.0
                      via so-0/3/0.0

inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
```

```
10.255.70.103/32   *[RSVP/7] 1d 13:20:13, metric 2
                    > via so-0/3/0.0, label-switched-path green-r1-r3

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
10.0.0.0/8         *[Direct/0] 2d 01:43:34
                    > via fxp2.0
                    [Direct/0] 2d 01:43:34
                    > via fxp1.0
```

show route best detail

```
user@host> show route best 10.255.70.103 detail
inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
10.255.70.103/32 (1 entry, 1 announced)
        *OSPF   Preference: 10
                Next-hop reference count: 9
                Next hop: 10.31.1.6 via ge-3/1/0.0, selected
                Next hop: via so-0/3/0.0
                State: <Active Int>
                Local AS:    69
                Age: 1d 13:20:06        Metric: 2
                Area: 0.0.0.0
                Task: OSPF
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
Restart Complete
10.255.70.103/32 (1 entry, 1 announced)
        State: <FlashAll>
        *RSVP   Preference: 7
                Next-hop reference count: 5
                Next hop: via so-0/3/0.0 weight 0x1, selected
                Label-switched-path green-r1-r3
                Label operation: Push 100016
                State: <Active Int>
                Local AS:    69
                Age: 1d 13:20:59        Metric: 2
                Task: RSVP
```

```
                Announcement bits (1): 1-Resolve tree 2
                AS path: I

private1__inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
10.0.0.0/8 (2 entries, 0 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 1
                Next hop: via fxp2.0, selected
                State: <Active Int>
                Age: 2d 1:44:20
                Task: IF
                AS path: I
         Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 1
                Next hop: via fxp1.0, selected
                State: <NotBest Int>
                Inactive reason: No difference
                Age: 2d 1:44:20
                Task: IF
                AS path: I
```

show route best extensive

The output for the `show route best extensive` command is identical to that for the `show route best detail` command. For sample output, see "show route best detail" on page 390.

show route best terse

```
user@host> show route best 10.255.70.103 terse
inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 10.255.70.103/32   O  10          2            >10.31.1.6
                                                  so-0/3/0.0

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

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

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 10.255.70.103/32   R   7          2            >so-0/3/0.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 10.0.0.0/8         D   0                       >fxp2.0
                     D   0                       >fxp1.0
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route brief

show route detail

show route extensive

show route terse

show route brief

IN THIS SECTION

Syntax

```
show route brief
<destination-prefix>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route brief
<destination-prefix>
```

Description

Display brief information about the active entries in the routing tables.

Options

none	Display all active entries in the routing table.
destination-prefix	(Optional) Display active entries for the specified address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

For information about output fields, see the Output Field table of the `show route` command.

Sample Output

show route brief

```
user@host> show route brief
inet.0: 10 destinations, 10 routes (9 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[Static/5] 1w5d 20:30:29
                      Discard
10.255.245.51/32   *[Direct/0] 2w4d 13:11:14
                    > via lo0.0
172.16.0.0/12      *[Static/5] 2w4d 13:11:14
                    > to 192.168.167.254 via fxp0.0
192.168.0.0/18         *[Static/5] 1w5d 20:30:29
                    > to 192.168.167.254 via fxp0.0
192.168.40.0/22    *[Static/5] 2w4d 13:11:14
                    > to 192.168.167.254 via fxp0.0
192.168.64.0/18    *[Static/5] 2w4d 13:11:14
                    > to 192.168.167.254 via fxp0.0
192.168.164.0/22   *[Direct/0] 2w4d 13:11:14
                    > via fxp0.0
192.168.164.51/32  *[Local/0] 2w4d 13:11:14
                      Local via fxp0.0
207.17.136.192/32  *[Static/5] 2w4d 13:11:14
                    > to 192.168.167.254 via fxp0.0
green.inet.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
100.101.0.0/16     *[Direct/0] 1w5d 20:30:28
```

```
                    > via fe-0/0/3.0
100.101.2.3/32     *[Local/0] 1w5d 20:30:28
                      Local via fe-0/0/3.0
172.16.233.5/32       *[OSPF/10] 1w5d 20:30:29, metric 1
                      MultiRecv
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route all

show route best

show route detail

IN THIS SECTION

Syntax

```
show route detail
<destination-prefix>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route detail
<destination-prefix>
```

Description

Display detailed information about the active entries in the routing tables.

Options

none	Display all active entries in the routing table on all systems.
destination-prefix	(Optional) Display active entries for the specified address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 15 on page 397 describes the output fields for the `show route detail` command. Output fields are listed in the approximate order in which they appear.

Table 16 on page 407 describes all possible values for the Next-hop Types output field.

Table 17 on page 409 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 18 on page 412 describes the possible values for the Communities output field.

Table 15: show route detail Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).
number destinations	Number of destinations for which there are routes in the routing table.
number routes	Number of routes in the routing table and total number of routes in the following states: • `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 15: show route detail Output Fields *(Continued)*

Field Name	Field Description

route-destination (entry, announced)	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:
	• *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).
	  • *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`.
	  • *source*—Source of the advertisement: `Local` or `Remote`.

label stacking	(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. • `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).

Table 15: show route detail Output Fields *(Continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th></tr>
<tr><td>[protocol, preference]</td><td>
Protocol from which the route was learned and the preference value for the route.

<code>+</code>—A plus sign indicates the active route, which is the route installed from the routing table into the forwarding table.
<code>-</code> —A hyphen indicates the last active route.
<code>*</code>—An asterisk indicates that the route is both the active and the last active route. An asterisk before a <code>to</code> line indicates the best subpath to the route.

In every routing metric except for the BGP <code>LocalPref</code> attribute, a lesser value is preferred. In order to use common comparison routines, Junos OS stores the 1's complement of the <code>LocalPref</code> value in the <code>Preference2</code> field. For example, if the <code>LocalPref</code> value for Route 1 is 100, the <code>Preference2</code> value is -101. If the <code>LocalPref</code> value for Route 2 is 155, the <code>Preference2</code> value is -156. Route 2 is preferred because it has a higher <code>LocalPref</code> value.
<code>Preference2</code> values are signed integers, that is, <code>Preference2</code> values can be either positive or negative values. However, Junos OS evaluates <code>Preference2</code> values as unsigned integers that are represented by positive values. Based on the <code>Preference2</code> values, Junos OS evaluates a preferred route differently in the following scenarios:

Both Signed <code>Preference2</code> values

Route A = -101
Route B = -156

Where both the <code>Preference2</code> values are signed, Junos OS evaluates only the unsigned value of <code>Preference2</code> and Route A, which has a lower <code>Preference2</code> value is preferred.
Unsigned <code>Preference2</code> values
Now consider both unsigned <code>Preference2</code> values:

Route A = 4294967096
Route B = 200

Here, Junos OS considers the lesser <code>Preference2</code> value and Route B with a <code>Preference2</code> value of 200 is preferred because it is less than 4294967096.
Combination of signed and unsigned <code>Preference2</code> values

</td></tr>
</table>

Table 15: show route detail Output Fields *(Continued)*

Field Name	Field Description
	When `Preference2` values of two routes are compared, and for one route the `Preference2` is a signed value, and for the other route it is an unsigned value, Junos OS prefers the route with the positive `Preference2` value over the negative `Preference2` value. For example, consider the following signed and unsigned `Preference2` values: • Route A = -200 • Route B = 200 In this case, Route B with a `Preference2` value of 200 is preferred although this value is greater than -200, because Junos OS evaluates only the unsigned value of the `Preference2` value.
Level	(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.
Route Distinguisher	IP subnet augmented with a 64-bit prefix.
PMSI	Provider multicast service interface (MVPN routing table).
Next-hop type	Type of next hop. For a description of possible values for this field, see Table 16 on page 407.
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.

Table 15: show route detail Output Fields *(Continued)*

Field Name	Field Description
Next hop	Network layer address of the directly reachable neighboring system.
via	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: • 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 17 on page 409.
Local AS	AS number of the local routing device.

Table 15: show route detail Output Fields *(Continued)*

Field Name	Field Description
Age	How long the route has been known.
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.
Metric*n*	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.
MED-plus-IGP	Metric value for BGP path selection to which the IGP cost to the next-hop destination has been added.
TTL-Action	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. For sample output, see `show route table` .
Task	Name of the protocol that has added the route.
Announcement bits	The number of BGP peers or protocols to which Junos OS has announced this route, followed by the list of the recipients of the announcement. Junos OS can also announce the route to the KRT for installing the route into the Packet Forwarding Engine, to a resolve tree, a L2 VC, or even a VPN. For example, `n-Resolve inet` indicates that the specified route is used for route resolution for next hops found in the routing table. • *n*—An index used by Juniper Networks customer support only.

Table 15: show route detail Output Fields *(Continued)*

Field Name	Field Description
AS path	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: • `I`—IGP. • `E`—EGP. • `Recorded`—The AS path is recorded by the sample process (sampled). • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—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. **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.

Table 15: show route detail Output Fields *(Continued)*

Field Name	Field Description
validation-state	(BGP-learned routes) Validation status of the route: • `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. • `Unverified`—Indicates that the origin of the prefix is not verified against the database. This is because the database got populated and the validation is not called for in the BGP import policy, although origin validation is enabled, or the origin validation is not enabled for the BGP peers. • `Valid`—Indicates that the prefix and autonomous system pair are found in the database.
ORR Generation-ID	Displays the optimal route reflection (ORR) generation identifier. ISIS and OSPF interior gateway protocol (IGP) updates filed whenever any of the corresponding ORR route has its metric valued changed, or if the ORR route is added or deleted.
FECs bound to route	Point-to-multipoint root address, multicast source address, and multicast group address when multipoint LDP (M-LDP) inband signaling is configured.
Primary Upstream	When multipoint LDP with multicast-only fast reroute (MoFRR) is configured, the primary upstream path. MoFRR transmits a multicast join message from a receiver toward a source on a primary path, while also transmitting a secondary multicast join message from the receiver toward the source on a backup path.
RPF Nexthops	When multipoint LDP with MoFRR is configured, the reverse-path forwarding (RPF) next-hop information. Data packets are received from both the primary path and the secondary paths. The redundant packets are discarded at topology merge points due to the RPF checks.
Label	Multiple MPLS labels are used to control MoFRR stream selection. Each label represents a separate route, but each references the same interface list check. Only the primary label is forwarded while all others are dropped. Multiple interfaces can receive packets using the same label.

Table 15: show route detail Output Fields *(Continued)*

Field Name	Field Description
weight	Value used to distinguish MoFRR primary and backup routes. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible.
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 18 on page 412 for all possible values for this field.
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).
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 LongLivedStale	The LongLivedStale flag indicates that the route was marked LLGR-stale by this router, as part of the operation of LLGR receiver mode. Either this flag or the LongLivedStaleImport flag may be displayed for a route. Neither of these flags are displayed at the same time as the Stale (ordinary GR stale) flag.

Table 15: show route detail Output Fields ***(Continued)***

Field Name	Field Description
Accepted LongLivedStaleImpor t	The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy. Either this flag or the LongLivedStale flag may be displayed for a route. Neither of these flags are displayed at the same time as the Stale (ordinary GR stale) flag. Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and import into the inet.0 routing table
ImportAccepted LongLivedStaleImpor t DeletePending	Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and imported into the inet.0 routing table The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy. The DeletePending flag indicates that a BGP route needs to be processed due to a BGP peer down event.
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.
Statistics ID Group	Indicates the Kernel ID number and Statistics ID number.
Statistics	Indicates the number of packets and data transferred.

Table 16: Next-hop Types Output Field Values

Next-Hop Type	Description
Broadcast (bcast)	Broadcast next hop.
Deny	Deny next hop.
Discard	Discard next hop.
Dynamic List	Dynamic list 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.
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.

Table 16: Next-hop Types Output Field Values *(Continued)*

Next-Hop Type	Description
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	A specific node or set of nodes to which the routing device forwards packets that match the route prefix. To qualify as next-hop type router, the route must meet the following criteria: • 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.
Software	Next hop added to the Routing Engine forwarding table for remote IP addresses with prefix /32 for Junos OS Evolved only.
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 17: 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.

Table 17: State Output Field Values *(Continued)*

Value	Description
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).
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.

Table 17: State Output Field Values *(Continued)*

Value	Description
NSR-incapable	Route added by non-NSR supported protocols.
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.
Programmed	Route installed programatically by on-box or off-box applications using API.
ProtectionCand	Indicates paths requesting protection.
ProtectionPath	Indicates the route entry that can be used as a protection path.
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.

Table 17: State Output Field Values *(Continued)*

Value	Description
Unusable path	Path is not usable because of one of the following conditions: • 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 18: Communities Output Field Values

Value	Description
area-number	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.
bandwidth: *local AS number:link-bandwidth-number*	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.
domain-id	Unique configurable number that identifies the OSPF domain.
domain-id-vendor	Unique configurable number that further identifies the OSPF domain.
link-bandwidth-number	Link-bandwidth number: from 0 through 4,294,967,295 (bytes per second).
local AS number	Local AS number: from 1 through 65,535.
options	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.

Table 18: Communities Output Field Values *(Continued)*

Value	Description
origin	(Used with VPNs) Identifies where the route came from.
ospf-route-type	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 be0); 7 for NSSA routes; or 129 for sham link endpoint addresses.
route-type-vendor	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*.
rte-type	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*.
target	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.
ipv6-origin	Defines the source of the IPv6 unicast address in a policy match condition.
ipv6-target	Defines the VPN IPv6 target unicast address used in a policy match condition. The target has the *128-bit IP address:16-bit number* format. For example, 2001:db8:9999::9.
ipv6-extended	Defines the extended format of the IPv6 unicast address in a policy match. For example, `ipv6-extended:203:<2001:db8::7>:67`
unknown IANA	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.
unknown OSPF vendor community	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:    65069
                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:    65069
                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:    65069
                Age: 1:30:17    Metric: 1
                ORR Generation-ID: 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:    65069
                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:    65069
                Age: 1:29:56    Metric: 2
                Area: 0.0.0.0
                ORR Generation-ID: 1
                        Task: OSPF
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

...

172.16.233.2/32 (1 entry, 1 announced)
        *PIM    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    65069
                Age: 1:31:45
                Task: PIM Recv
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

...

172.16.233.22/32 (1 entry, 1 announced)
```

```
        *IGMP   Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    65069
                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:    65069
                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:    65069
                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:    65069
                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:    65069
                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)

  299840 (1 entry, 1 announced)
  TSI:
  KRT in-kernel 299840 /52 -> {indirect(1048575)}
          *RSVP   Preference: 7/2
                  Next hop type: Flood
                  Address: 0x9174a30
                  Next-hop reference count: 4
                  Next hop type: Router, Next hop index: 798
                  Address: 0x9174c28
                  Next-hop reference count: 2
                  Next hop: 172.16.0.2 via lt-1/2/0.9 weight 0x1
                  Label-switched-path R2-to-R4-2p2mp
                  Label operation: Pop
                  Next hop type: Router, Next hop index: 1048574
                  Address: 0x92544f0
                  Next-hop reference count: 2
                  Next hop: 172.16.0.2 via lt-1/2/0.7 weight 0x1
```

```
                  Label-switched-path R2-to-R200-p2mp
                  Label operation: Pop
                  Next hop: 172.16.0.2 via lt-1/2/0.5 weight 0x8001
                  Label operation: Pop
                  State: <Active Int>
                  Age: 1:29       Metric: 1
                  Task: RSVP
                  Announcement bits (1): 0-KRT
                  AS path: I...

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:    65069
                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:    65069
                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:    65069
                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:    65069
                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:    65069
                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:    65069 Peer AS:    65069
                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

...

l2circuit.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:    65099
                Age: 10:21
                Task: l2 circuit
                Announcement bits (1): 0-LDP
                AS path: I
                VC Label 100000, MTU 1500, VLAN ID 512

inet.0: 45 destinations, 47 routes (44 active, 0 holddown, 1 hidden)
```

```
10.1.1.3/32 (1 entry, 1 announced)
        *IS-IS  Preference: 18
                Level: 2
                Next hop type: Router, Next hop index: 580
                Address: 0x9db6ed0
                Next-hop reference count: 8
                Next hop: 10.1.1.6 via lt-1/0/10.5, selected
                Session Id: 0x18a
                State: <Active Int>
                Local AS:     2
                Age: 1:32       Metric: 10
                Validation State: unverified
                ORR Generation-ID: 1
                Task: IS-IS
                Announcement bits (3): 0-KRT 5-Resolve tree 4 6-Resolve_IGP_FRR task
                AS path: I

inet.0: 61 destinations, 77 routes (61 active, 1 holddown, 0 hidden)
10.1.1.1/32 (2 entries, 1 announced)
        *OSPF   Preference: 10
                Next hop type: Router, Next hop index: 673
                Address: 0xc008830
                Next-hop reference count: 3
                Next hop: 10.1.1.8 via ge-0/0/2.0, selected
                Session Id: 0x1b7
                State: <Active Int>
                Local AS:     1
                Age: 3:06:59    Metric: 100
                Validation State: unverified
                ORR Generation-ID: 1
                Area: 0.0.0.0
                Task: OSPF
                Announcement bits (2): 1-KRT 9-Resolve tree 2
                AS path: I
```

show route programmed detail (with statistics information)

```
user@host> show route programmed detail

inet.0: 104 destinations, 104 routes (103 active, 0 holddown, 1 hidden)
```

```
inet.3: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
192.0.2.0/24 (1 entry, 1 announced)
        *Static Preference: 5/100
                Next hop type: Router, Next hop index: 0
                Address: 0xa667694
                Next-hop reference count: 15
                Next hop: 198.51.100.0 via ae0.0
                Label operation: Push 50008, Push 3400(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3400: None;
                Label element ptr: 0x84e7e90
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 10, Stats IDs = { 536870923 }
                Statistics: Packets 1380, Bytes 681720
                Next hop: 198.51.100.1 via ae0.1
                Label operation: Push 50008, Push 3401(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3401: None;
                Label element ptr: 0x84e7ee0
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 11, Stats IDs = { 536870924 }
                Statistics: Packets 1444, Bytes 713336
                Next hop: 198.51.100.2 via ae0.2
                Label operation: Push 50008, Push 3410(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3410: None;
                Label element ptr: 0xa1926a0
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 12, Stats IDs = { 536870925 }
                Statistics: Packets 1420, Bytes 701480
                Next hop: 198.51.100.3 via ge-0/0/2.0
```

```
Label operation: Push 50008, Push 3411(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3411: None;
Label element ptr: 0xa1926c8
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 13, Stats IDs = { 536870926 }
Statistics: Packets 1486, Bytes 734084
Next hop: 198.51.100.4 via ge-0/0/2.1, selected
Label operation: Push 50008, Push 3420(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3420: None;
Label element ptr: 0xa1926f0
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 14, Stats IDs = { 536870927 }
Statistics: Packets 1399, Bytes 691106
Next hop: 198.51.100.5 via ge-0/0/2.2
Label operation: Push 50008, Push 3421(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3421: None;
Label element ptr: 0xa192718
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 15, Stats IDs = { 536870928 }
Statistics: Packets 1455, Bytes 718770
Next hop: 198.51.100.6 via ge-0/0/4.0
Label operation: Push 50008, Push 3450(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3450: None;
Label element ptr: 0xa192740
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
```

```
                  Label element lsp id: 0
                  Session Id: 0x0
                  Statistics ID Group: Kernel ID = 16, Stats IDs = { 536870929 }
                  Statistics: Packets 1407, Bytes 695058
                  Next hop: 198.51.100.7 via ge-0/0/4.1
                  Label operation: Push 50008, Push 3460(top)
                  Label TTL action: prop-ttl, prop-ttl(top)
                  Load balance label: Label 50008: None; Label 3460: None;
                  Label element ptr: 0xa192768
                  Label parent element ptr: 0x0
                  Label element references: 3
                  Label element child references: 0
                  Label element lsp id: 0
                  Session Id: 0x0
                  Statistics ID Group: Kernel ID = 17, Stats IDs = { 536870930 }
                  Statistics: Packets 1418, Bytes 700492
                  Next hop: 198.51.100.8 via ge-0/0/4.2
                  Label operation: Push 50008, Push 3470(top)
                  Label TTL action: prop-ttl, prop-ttl(top)
                  Load balance label: Label 50008: None; Label 3470: None;
                  Label element ptr: 0xa192790
                  Label parent element ptr: 0x0
                  Label element references: 3
                  Label element child references: 0
                  Label element lsp id: 0
                  Session Id: 0x0
                  Statistics ID Group: Kernel ID = 18, Stats IDs = { 536870931 }
                  Statistics: Packets 0, Bytes 0
                  State: <Active Int NSR-incapable Programmed>
                  Age: 12:46
                  Validation State: unverified
                  Announcement bits (3): 0-Resolve tree 6 1-Resolve tree 1 3-Resolve_IGP_FRR task
                  AS path: I
                  Session-IDs associated:
                  Session-id: 370 Version: 0
                  Thread: junos-main
```

show route detail (with IPv6 unicast address community)

```
user@host> show route 203.0.112.1 detail
inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
```

```
203.0.112.1/32 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 559
                Address: 0x7b5de84
                Next-hop reference count: 12, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 13.1.1.1
                Next hop: 13.1.1.1 via ge-0/0/0.0, selected
                Session Id: 320
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 8:41       Metric: 1
                Validation State: unverified
                Task: BGP_100.13.1.1.1
                Announcement bits (1): 0-KRT
                AS path: 100 I
                Communities: target:100:65 target:172.16.7.7:2 ipv6-target:<2001:db8::5>:65
                ipv6-target:<2001:db8::5>:101 ipv6-origin:<2001:db8::6>:66
                ipv6-origin:<2001:db8::6>:102 ipv6-extended:100:<2001:db8::7>:67
                ipv6-extended:256:<2001:db8::7>:103
                Accepted
                Localpref: 100
                Router ID: 2.2.2.2
                Thread: junos-main
```

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
                  Validation State: unverified
                Task: BGP_2.10.1.1.2+59955
```

```
                Announcement bits (1): 0-KRT
                AS path: 65002 I
                Accepted Multipath
                Localpref: 100
                Router ID: 172.16.1.2
         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:     65001 Peer AS:     65002
                Age: 5:04:43
                   Validation State: unverified
                Task: BGP_2.10.1.1.6+58198
                AS path: 65002 I
                Accepted MultipathContrib
                Localpref: 100
                Router ID: 172.16.1.3
```

show route detail (with BGP, DeletePending)

```
user@host> show route detail
2001:db8:2:1:10.1.1.12/30 (1 entry, 0 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 65002:1
                Next hop type: Indirect
                Address: 0x95c4ee8
                Next-hop reference count: 6
                Source: 10.1.1.4
                Next hop type: Router, Next hop index: 809
                Next hop: 10.1.1.6 via lt-1/0/10.5, selected
                Label operation: Push 299888, Push 299792(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 299888: None; Label 299792: None;
                Session Id: 0x142
                Protocol next hop: 10.1.1.4
                Label operation: Push 299888
                Label TTL action: prop-ttl
```

```
                Load balance label: Label 299888: None;
                Indirect next hop: 0x96f0110 1048574 INH Session ID: 0x14e
                State: <Active Int Ext ProtectionPath ProtectionCand>
                Local AS:     65002 Peer AS:     65002
                Age: 2w1d 17:42:45      Metric2: 1
                Validation State: unverified
                Task: BGP_10.2.1.1.4+55190
                AS path: I
                Communities: target:65002:1
                Import Accepted DeletePending
                VPN Label: 299888
                Localpref: 100
                Router ID: 10.1.1.4
                Secondary Tables: red.inet.0
```

show route label detail (Multipoint LDP Inband Signaling for Point-to-Multipoint LSPs)

```
user@host> show route label 299872 detail
mpls.0: 13 destinations, 13 routes (13 active, 0 holddown, 0 hidden)
299872 (1 entry, 1 announced)
        *LDP    Preference: 9
                Next hop type: Flood
                Next-hop reference count: 3
                Address: 0x9097d90
                Next hop: via vt-0/1/0.1
                Next-hop index: 661
                Label operation: Pop
                Address: 0x9172130
                Next hop: via so-0/0/3.0
                Next-hop index: 654
                Label operation: Swap 299872
                State: **Active Int>
                Local AS:  1001
                Age: 8:20       Metric: 1
                Task: LDP
                Announcement bits (1): 0-KRT
                AS path: I
                FECs bound to route: P2MP root-addr 10.255.72.166, grp 232.1.1.1, src
192.168.142.2
```

show route label detail (Multipoint LDP with Multicast-Only Fast Reroute)

```
user@host> show route label 301568 detail

mpls.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
301568 (1 entry, 1 announced)
        *LDP    Preference: 9
                Next hop type: Flood
                Address: 0x2735208
                Next-hop reference count: 3
                Next hop type: Router, Next hop index: 1397
                Address: 0x2735d2c
                Next-hop reference count: 3
                Next hop: 1.3.8.2 via ge-1/2/22.0
                Label operation: Pop
                Load balance label:  None;
                Next hop type: Router, Next hop index: 1395
                Address: 0x2736290
                Next-hop reference count: 3
                Next hop: 1.3.4.2 via ge-1/2/18.0
                Label operation: Pop
                Load balance label:  None;
                State: <Active Int AckRequest MulticastRPF>
                Local AS:    10
                Age: 54:05      Metric: 1
                Validation State: unverified
                Task: LDP
                Announcement bits (1): 0-KRT
                AS path: I
                FECs bound to route: P2MP root-addr 172.16.1.1, grp: 232.1.1.1, src:
192.168.219.11
                Primary Upstream : 172.16.1.3:0--172.16.1.2:0
                   RPF Nexthops  :
                        ge-1/2/15.0, 10.1.94.1, Label: 301568, weight: 0x1
                        ge-1/2/14.0, 10.1.3.1, Label: 301568, weight: 0x1
                Backup Upstream : 172.16.1.3:0--172.16.1.6:0
                   RPF Nexthops :
                        ge-1/2/20.0, 10.1.96.1, Label: 301584, weight: 0xfffe
                        ge-1/2/19.0, 10.1.6.1, Label: 301584, weight: 0xfffe
```

show route detail (Flexible VXLAN Tunnel Profile)

```
user@host> show route 192.168.0.2 detail
...

CUSTOMER_0001.inet.0: 5618 destinations, 6018 routes (5618 active, 0 holddown, 0 hidden)

192.168.0.2/32 (1 entry, 1 announced)
        *Static Preference: 5/100
                Next hop type: Router, Next hop index: 74781
                Address: 0x5d9b03cc
                Next-hop reference count: 363
                Next hop: via fti0.6, selected
                Session Id: 0x24c8
                State: <Active Int NSR-incapable OpaqueData Programmed>
                Age: 1:25:53
                Validation State: unverified
                        Tag: 10000001   Tag2: 1
                Announcement bits (2): 1-KRT 3-Resolve tree 30
                AS path: I
                Flexible IPv6 VXLAN tunnel profile
                    Action: Encapsulate
                    Interface: fti0.6 (Index: 10921)
                    VNI: 10000001
                    Source Prefix: 2001:db8:255::2/128
                    Source UDP Port Range: 54614 - 60074
                    Source MAC Address: 00:00:5e:00:52:01
                    Destination Address: 2001:db8:10:1:1:1:0:1
                    Destination UDP Port: 4790
                    VXLAN Flags: 0x08
...
```

Release Information

Command introduced before Junos OS Release 7.4.

Comand introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

`DeletePending` flag added to the command output in Junos OS Release 19.4R1.

`ipv6-origin`, `ipv6-target`, and `ipv6-extended Community` output fields supported in Junos OS Release and Junos OS Evolved Release 23.1.

show route exact

IN THIS SECTION

Syntax

```
show route exact destination-prefix
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route exact destination-prefix
<brief | detail | extensive | terse>
```

Description

Display only the routes that exactly match the specified address or range of addresses.

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`.
destination-prefix	Address or range of addresses.
logical-system (all | ***logical-system-name*****)**	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 exact

```
user@host> show route exact 207.17.136.0/24

inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
```

```
207.17.136.0/24    *[Static/5] 2d 03:30:22
                    > to 192.168.71.254 via fxp0.0
```

show route exact detail

```
user@host> show route exact 207.17.136.0/24 detail

inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
207.17.136.0/24 (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: 2d 3:30:26
                Task: RT
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I
```

show route exact extensive

```
user@host> show route exact 207.17.136.0/24 extensive
inet.0: 22 destinations, 23 routes (21 active, 0 holddown, 1 hidden)
207.17.136.0/24 (1 entry, 1 announced)
TSI:
KRT in-kernel 207.17.136.0/24 -> {192.168.71.254}
        *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: 1:25:18
                Task: RT
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I
```

show route exact terse

```
user@host> show route exact 207.17.136.0/24 terse

inet.0: 22 destinations, 23 routes (21 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 207.17.136.0/24    S   5                       >192.168.71.254
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route export

IN THIS SECTION

Syntax

```
show route export
<brief | detail>
<instance <instance-name> | routing-table-name>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route export
<brief | detail>
<instance <instance-name> | routing-table-name>
```

Description

Display policy-based route export information. Policy-based export simplifies the process of exchanging route information between routing instances.

Options

none	(Same as `brief`.) Display standard information about policy-based export for all instances and routing tables on all systems.
brief | detail	(Optional) Display the specified level of output.

instance <*instance-name*>	(Optional) Display a particular routing instance for which policy-based export is currently enabled.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
routing-table-name	(Optional) Display information about policy-based export 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 export inet` command).

Required Privilege Level

view

Output Fields

Table 19 on page 436 lists the output fields for the `show route export` command. Output fields are listed in the approximate order in which they appear.

Table 19: show route export Output Fields

Field Name	Field Description	Level of Output
Table or *table-name*	Name of the routing tables that either import or export routes.	All levels
Routes	Number of routes exported from this table into other tables. If a particular route is exported to different tables, the counter will only increment by one.	`brief` none
Export	Whether the table is currently exporting routes to other tables: Y or N (Yes or No).	`brief` none
Import	Tables currently importing routes from the originator table. (Not displayed for tables that are not exporting any routes.)	`detail`

Table 19: show route export Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Flags	(`instance` keyword only) Flags for this feature on this instance: • `config auto-policy`—The policy was deduced from the configured IGP export policies. • `cleanup`—Configuration information for this instance is no longer valid. • `config`—The instance was explicitly configured.	`detail`
Options	(`instance` keyword only) Configured option displays the type of routing tables the feature handles: • `unicast`—Indicates *instance*.inet.0. • `multicast`—Indicates *instance*.inet.2. • `unicast multicast`—Indicates *instance*.inet.0 and *instance*.inet.2.	`detail`
Import policy	(`instance` keyword only) Policy that `route export` uses to construct the import-export matrix. Not displayed if the instance type is `vrf`.	`detail`
Instance	(`instance` keyword only) Name of the routing instance.	`detail`
Type	(`instance` keyword only) Type of routing instance: `forwarding`, `non-forwarding`, or `vrf`.	`detail`

Sample Output

show route export

```
user@host> show route export
Table                            Export          Routes
inet.0                           N                    0
```

```
black.inet.0                    Y                     3
red.inet.0                      Y                     4
```

show route export detail

```
user@host> show route export detail
inet.0                                          Routes:        0
black.inet.0                                    Routes:        3
  Import: [ inet.0 ]
red.inet.0                                      Routes:        4
  Import: [ inet.0 ]
```

show route export instance detail

```
user@host> show route export instance detail
Instance: master                        Type: forwarding
  Flags: <config auto-policy> Options: <unicast multicast>
  Import policy: [ (ospf-master-from-red || isis-master-from-black) ]
Instance: black                         Type: non-forwarding
Instance: red                           Type: non-forwarding
```

Release Information

Command introduced before Junos OS Release 7.4.

show route extensive

IN THIS SECTION

Syntax

```
show route extensive
<destination-prefix>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route extensive
<destination-prefix>
```

Description

Display extensive information about the active entries in the routing tables.

Options

none	Display all active entries in the routing table.

destination-prefix	(Optional) Display active entries for the specified address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 20 on page 440 describes the output fields for the `show route extensive` command. Output fields are listed in the approximate order in which they appear.

Table 20: show route extensive Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).
number destinations	Number of destinations for which there are routes in the routing table.
number routes	Number of routes in the routing table and total number of routes in the following states: • `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 20: show route extensive Output Fields *(Continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th></tr>
<tr><td>route-destination (entry, announced)</td><td>Route destination (for example: 10.0.0.1/24). The <code>entry</code> value is the number of route for this destination, and the <code>announced</code> value is the number of routes being announced for this destination. Sometimes the route destination is presented in another format, such as:

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).

neighbor-address—Address of the neighbor.
control-word-status—Whether the use of the control word has been negotiated for this virtual circuit: <code>NoCtrlWord</code> or <code>CtrlWord</code>.
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: <code>Local</code> or <code>Remote</code>.

</td></tr>
<tr><td>TSI</td><td>Protocol header information.</td></tr>
<tr><td>label stacking</td><td>(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.

<code>S=0 route</code> indicates that a packet with an incoming label stack depth of two or more exits this router with one fewer label (the label-popping operation is performed).
If there is no <code>S=</code> information, the route is a normal MPLS route, which has a stack depth of 1 (the label-popping operation is not performed).
</td></tr>
</table>

Table 20: show route extensive Output Fields *(Continued)*

Field Name	Field Description
[*protocol*, *preference*]	Protocol from which the route was learned and the preference value for the route. • `+`—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. 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.
Level	(IS-IS only). In IS-IS, a single autonomous system (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.
Route Distinguisher	IP subnet augmented with a 64-bit prefix.
PMSI	Provider multicast service interface (MVPN routing table).
Next-hop type	Type of next hop.
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.

Table 20: show route extensive Output Fields *(Continued)*

Field Name	Field Description
Source	IP address of the route source.
Next hop	Network layer address of the directly reachable neighboring system.
via	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: • 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).
Offset	Whether the metric has been increased or decreased by an offset value.
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 recursively derive a forwarding 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).

Table 20: show route extensive Output Fields *(Continued)*

Field Name	Field Description
Indirect next hops	When present, a list of nodes that are used to resolve the path to the next-hop destination, in the order that they are resolved. When BGP PIC Edge is enabled, the output lines that contain `Indirect next hop: weight` follow next hops that the software can use to repair paths where a link failure occurs. The next-hop weight has one of the following values: • 0x1 indicates active next hops. • 0x4000 indicates passive next hops.
State	State of the route (a route can be in more than one state).
Session ID	The BFD session ID number that represents the protection using MPLS fast reroute (FRR) and loop-free alternate (LFA).
Weight	Weight for the backup path. If the weight of an indirect next hop is larger than zero, the weight value is shown.

Table 20: show route extensive Output Fields *(Continued)*

Field Name	Field Description
Inactive reason	If the route is inactive, the reason for its current state is indicated. Typical reasons include: • `Active preferred`—Currently active route was selected over this route. • `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. • `Cluster list length`—Path with a shorter cluster list length is available. • `Forwarding use only`—Path is only available for forwarding purposes. • `IGP metric`—Path through the next hop with a lower IGP metric is available. • `IGP metric type`—Path with a lower OSPF link-state advertisement type is available. • `Interior > Exterior > Exterior via Interior`—Direct, static, IGP, or EBGP path is available. • `Local preference`—Path with a higher local preference value is available. • `Next hop address`—Path with a lower metric next hop is available. • `No difference`—Paths from all neighbors are identical for all necessary route attributes in some special scenarios, such as BGP L2VPN/VPLS, and there is no difference. • `Not Best in its group`—Occurs when multiple peers of the same external AS advertise the same prefix and are grouped together in the selection process. When this reason is displayed, an additional reason is provided (typically one of the other reasons listed). • `Number of gateways`—Path with a higher number of next hops is available. • `Origin`—Path with a lower origin code is available. • `OSPF version`—Path does not support the indicated OSPF version. • `RIB preference`—Route from a higher-numbered routing table is available. • `Route destinguisher`—64-bit prefix added to IP subnets to make them unique.

Table 20: show route extensive Output Fields *(Continued)*

Field Name	Field Description
	• `Route metric or MED comparison`—Route with a lower metric or MED is available. • `Route preference`—Route with a lower preference value is available. • `Router ID`—Path through a neighbor with a lower ID is available. • `Unusable path`—Path is not usable because of one of the following conditions: the route is damped, the route is rejected by an import policy, or the route is unresolved. • `Update source`—Last tiebreaker is the lowest IP address value.
Local AS	Autonomous system (AS) number of the local routing device.
Age	How long the route has been known.
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.
Metric	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.
MED-plus-IGP	Metric value for BGP path selection to which the IGP cost to the next-hop destination has been added.
TTL-Action	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.
Task	Name of the protocol that has added the route.

Table 20: show route extensive Output Fields *(Continued)*

Field Name	Field Description
Announcement bits	List of protocols that are consumers of the route. Using the following output as an example, `Announcement bits (3): 0-KRT 5-Resolve tree 2 8-BGP RT Background` there are (3) announcement bits to reflect the three clients (protocols) that have state for this route: Kernel (0-KRT), 5 (resolution tree process 2), and 8 (BGP). The notation *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	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: • `I`—IGP. • `E`—EGP. • `Recorded`—The AS path is recorded by the sample process (sampled). • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—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. **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.

Table 20: show route extensive Output Fields *(Continued)*

Field Name	Field Description
validation-state	(BGP-learned routes) Validation status of the route: • `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. • `Unverified`—Indicates that origin validation is not enabled for the BGP peers. • `Valid`—Indicates that the prefix and autonomous system pair are found in the database.
FECs bound to route	Point-to-multipoint root address, multicast source address, and multicast group address when multipoint LDP (M-LDP) inband signaling is configured.
AS path: I <Originator>	(For route reflected output only) Originator ID attribute set by the route reflector.
route status	Indicates the status of a BGP route: • `Accepted`—The specified BGP route is imported by the default BGP policy. • `Import`—The route is imported into a Layer 3 VPN routing instance. • `Import-Protect`—A remote instance egress that is protected. • `Multipath`—A BGP multipath active route. • `MultipathContrib`—The route is not active but contributes to the BGP multipath. • `Protect`—An egress route that is protected. • `Stale`—A route that is marked stale due to graceful restart.
Primary Upstream	When multipoint LDP with multicast-only fast reroute (MoFRR) is configured, the primary upstream path. MoFRR transmits a multicast join message from a receiver toward a source on a primary path, while also transmitting a secondary multicast join message from the receiver toward the source on a backup path.

Table 20: show route extensive Output Fields *(Continued)*

Field Name	Field Description
RPF Nexthops	When multipoint LDP with MoFRR is configured, the reverse-path forwarding (RPF) next-hop information. Data packets are received from both the primary path and the secondary paths. The redundant packets are discarded at topology merge points due to the RPF checks.
Label	Multiple MPLS labels are used to control MoFRR stream selection. Each label represents a separate route, but each references the same interface list check. Only the primary label is forwarded while all others are dropped. Multiple interfaces can receive packets using the same label.
weight	Value used to distinguish MoFRR primary and backup routes. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible.
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.
Cluster list	(For route reflected output only) Cluster ID sent by the route reflector.
Originator ID	(For route reflected output only) Address of router that originally sent the route to the route reflector.
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.
DeletePending	The DeletePending flag indicates that a BGP route needs to be processed due to a BGP peer down event.
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).

Table 20: show route extensive 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).
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.
Originating RIB	Name of the routing table whose active route was used to determine the forwarding next-hop entry in the resolution database. For example, in the case of inet.0 resolving through inet.0 and inet.3, this field indicates which routing table, inet.0 or inet.3, provided the best path for a particular prefix.
Node path count	Number of nodes in the path.
Forwarding nexthops	Number of forwarding next hops. The forwarding next hop is the network layer address of the directly reachable neighboring system (if applicable) and the interface used to reach it.
Statistics ID Group	Indicates the Kernel ID number and Statistics ID number.
Statistics	Indicates the number of packets and data transferred.

Sample Output

show route extensive

```
user@host> show route extensive
inet.0: 22 destinations, 23 routes (21 active, 0 holddown, 1 hidden)
203.0.113.10/16 (1 entry, 1 announced)
TSI:
KRT in-kernel 203.0.113.10/16 -> {192.168.71.254}
        *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:    64496
                Age: 1:34:06
                Task: RT
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

203.0.113.30/30 (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:    64496
                Age: 1:32:40
                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:    64496
                Age: 1:32:40    Metric: 1
                Area: 0.0.0.0
                Task: OSPF
                AS path: I
```

```
203.0.113.103/32 (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:    644969
                Age: 1:32:43
                Task: IF
                Announcement bits (1): 3-Resolve tree 2
                AS path: I

...

203.0.113.203/30 (1 entry, 1 announced)
TSI:
KRT in-kernel 203.0.113.203/30 -> {203.0.113.216}
        *OSPF   Preference: 10
                Next-hop reference count: 9
                Next hop: via so-0/3/0.0
                Next hop: 203.0.113.216 via ge-3/1/0.0, selected
                State: <Active Int>
                Local AS:    64496
                Age: 1:32:19    Metric: 2
                Area: 0.0.0.0
                Task: OSPF
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

...

198.51.100.2/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 198.51.100.2/32 -> {}
        *PIM    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:08
                Task: PIM Recv
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I
```

```
...

198.51.100.22/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 198.51.100.22/32 -> {}
        *IGMP   Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:06
                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)

203.0.113.103/32 (1 entry, 1 announced)
        State: <FlashAll>
        *RSVP   Preference: 7
                Next-hop reference count: 6
                Next hop: 203.0.113.216 via ge-3/1/0.0 weight 0x1, selected
                Label-switched-path green-r1-r3
                Label operation: Push 100096
                State: <Active Int>
                Local AS:    64496
                Age: 1:28:12    Metric: 2
                Task: RSVP
                Announcement bits (2): 1-Resolve tree 1 2-Resolve tree 2
                AS path: I

203.0.113.238/32 (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:    64496
                Age: 1:28:12    Metric: 1
                Task: RSVP
                Announcement bits (2): 1-Resolve tree 1 2-Resolve tree 2
                AS path: I
```

```
private1__.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:    64496
                Age: 1:34:07
                Task: IF
                AS path: I

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

0   (1 entry, 1 announced)
TSI:
KRT in-kernel 0      /36 -> {}
        *MPLS   Preference: 0
                Next hop type: Receive
                Next-hop reference count: 6
                State: <Active Int>
                Local AS:    64496
                Age: 1:34:08    Metric: 1
                Task: MPLS
                Announcement bits (1): 0-KRT
                AS path: I

...

mpls.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
  299840 (1 entry, 1 announced)
  TSI:
  KRT in-kernel 299840 /52 -> {indirect(1048575)}
          *RSVP   Preference: 7/2
                  Next hop type: Flood
                  Address: 0x9174a30
                  Next-hop reference count: 4
                  Next hop type: Router, Next hop index: 798
```

```
                    Address: 0x9174c28
                    Next-hop reference count: 2
                    Next hop: 198.51.100.2 via lt-1/2/0.9 weight 0x1
                    Label-switched-path R2-to-R4-2p2mp
                    Label operation: Pop
                    Next hop type: Router, Next hop index: 1048574
                    Address: 0x92544f0
                    Next-hop reference count: 2
                    Next hop: 198.51.100.2 via lt-1/2/0.7 weight 0x1
                    Label-switched-path R2-to-R200-p2mp
                    Label operation: Pop
                    Next hop: 198.51.100.2 via lt-1/2/0.5 weight 0x8001
                    Label operation: Pop
                    State: <Active Int>
                    Age: 1:29       Metric: 1
                    Task: RSVP
                    Announcement bits (1): 0-KRT
                    AS path: I...

800010 (1 entry, 1 announced)

TSI:
KRT in-kernel 800010 /36 -> {vt-3/2/0.32769}
        *VPLS   Preference: 7
                Next-hop reference count: 2
                Next hop: via vt-3/2/0.32769, selected
                Label operation: Pop
                State: <Active Int>
                Age: 1:31:53
                Task: Common L2 VC
                Announcement bits (1): 0-KRT
                AS path: I

vt-3/2/0.32769 (1 entry, 1 announced)
TSI:
KRT in-kernel vt-3/2/0.32769.0      /16 -> {indirect(1048574)}
        *VPLS   Preference: 7
                Next-hop reference count: 2
                Next hop: 203.0.113.216 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: 203.0.113.103
                Push 800012
```

```
                Indirect next hop: 87272e4 1048574
                State: <Active Int>
                Age: 1:31:53    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
                Indirect next hops: 1
                        Protocol next hop: 203.0.113.103 Metric: 2
                        Push 800012
                        Indirect next hop: 87272e4 1048574
                        Indirect path forwarding next hops: 1
                                Next hop: 203.0.113.216 via ge-3/1/0.0 weight 0x1
                        203.0.113.103/32 Originating RIB: inet.3
                          Metric: 2                       Node path count: 1
                          Forwarding nexthops: 1
                                Nexthop: 203.0.113.216 via ge-3/1/0.0

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

2001:db8::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:    64496
                Age: 1:34:07
                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:    64496
                Age: 1:34:07
                Task: IF
                AS path: I
```

```
ff02::2/128 (1 entry, 1 announced)
TSI:
KRT in-kernel ff02::2/128 -> {}
        *PIM    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:08
                Task: PIM Recv6
                Announcement bits (1): 0-KRT
                AS path: I

ff02::d/128 (1 entry, 1 announced)
TSI:
KRT in-kernel ff02::d/128 -> {}
        *PIM    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:08
                Task: PIM Recv6
                Announcement bits (1): 0-KRT
                AS path: I

ff02::16/128 (1 entry, 1 announced)
TSI:
KRT in-kernel ff02::16/128 -> {}
        *MLD    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:06
                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: 1:34:07
                Task: IF
                AS path: I

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

203.0.113.103:1:3:1/96 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 203.0.113.103:1
                Next-hop reference count: 7
                Source: 203.0.113.103
                Protocol next hop: 203.0.113.103
                Indirect next hop: 2 no-forward
                State: <Secondary Active Int Ext>
                Local AS:    64496 Peer AS:    64496
                Age: 1:28:12    Metric2: 1
                Task: BGP_69.203.0.113.103+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: 203.0.113.103
                Primary Routing Table bgp.l2vpn.0

203.0.113.152:1:1:1/96 (1 entry, 1 announced)
TSI:
Page 0 idx 0 Type 1 val 8699540
        *L2VPN  Preference: 170/-1
                Next-hop reference count: 5
                Protocol next hop: 203.0.113.152
                Indirect next hop: 0 -
                State: <Active Int Ext>
                Age: 1:34:03    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
```

```
203.0.113.152:1:5:1/96 (1 entry, 1 announced)
TSI:
Page 0 idx 0 Type 1 val 8699528
        *L2VPN  Preference: 170/-101
                Next-hop reference count: 5
                Protocol next hop: 203.0.113.152
                Indirect next hop: 0 -
                State: <Active Int Ext>
                Age: 1:34:03    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

...

l2circuit.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

TSI:

203.0.113.163: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: 203.0.113.163 Indirect next hop: 86af000 296
                State: <Active Int>
                Local AS:    64499
                Age: 10:21
                Task: l2 circuit
                Announcement bits (1): 0-LDP
                AS path: I
                VC Label 100000, MTU 1500, VLAN ID 512



203.0.113.55/24 (1 entry, 1 announced)
TSI:
KRT queued (pending) add
  198.51.100.0/24 -> {Push 300112}
        *BGP    Preference: 170/-101
                Next hop type: Router
                Address: 0x925c208
```

```
                Next-hop reference count: 2
                Source: 203.0.113.9
                Next hop: 203.0.113.9 via ge-1/2/0.15, selected
                Label operation: Push 300112
                Label TTL action: prop-ttl
                State: <Active Ext>
                Local AS:  64509 Peer AS: 65539
                Age: 1w0d 23:06:56
                AIGP: 25
                Task: BGP_65539.203.0.113.9+56732
                Announcement bits (1): 0-KRT
                AS path: 65539 64508 I
                Accepted
                Route Label: 300112
                Localpref: 100
                Router ID: 213.0.113.99
```

show route programmed extensive (with statistics information)

```
user@host> show route programmed extensive

inet.0: 104 destinations, 104 routes (103 active, 0 holddown, 1 hidden)

inet.3: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
192.0.2.0/24 (1 entry, 1 announced)
        *Static Preference: 5/100
                Next hop type: Router, Next hop index: 0
                Address: 0xa667694
                Next-hop reference count: 15
                Next hop: 198.51.100.0 via ae0.0
                Label operation: Push 50008, Push 3400(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3400: None;
                Label element ptr: 0x84e7e90
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 10, Stats IDs = { 536870923 }
                Statistics: Packets 1380, Bytes 681720
```

```
Next hop: 198.51.100.1 via ae0.1
Label operation: Push 50008, Push 3401(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3401: None;
Label element ptr: 0x84e7ee0
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 11, Stats IDs = { 536870924 }
Statistics: Packets 1444, Bytes 713336
Next hop: 198.51.100.2 via ae0.2
Label operation: Push 50008, Push 3410(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3410: None;
Label element ptr: 0xa1926a0
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 12, Stats IDs = { 536870925 }
Statistics: Packets 1420, Bytes 701480
Next hop: 198.51.100.3 via ge-0/0/2.0
Label operation: Push 50008, Push 3411(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3411: None;
Label element ptr: 0xa1926c8
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 13, Stats IDs = { 536870926 }
Statistics: Packets 1486, Bytes 734084
Next hop: 198.51.100.4 via ge-0/0/2.1, selected
Label operation: Push 50008, Push 3420(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3420: None;
Label element ptr: 0xa1926f0
Label parent element ptr: 0x0
Label element references: 3
```

```
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 14, Stats IDs = { 536870927 }
Statistics: Packets 1399, Bytes 691106
Next hop: 198.51.100.5 via ge-0/0/2.2
Label operation: Push 50008, Push 3421(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3421: None;
Label element ptr: 0xa192718
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 15, Stats IDs = { 536870928 }
Statistics: Packets 1455, Bytes 718770
Next hop: 198.51.100.6 via ge-0/0/4.0
Label operation: Push 50008, Push 3450(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3450: None;
Label element ptr: 0xa192740
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 16, Stats IDs = { 536870929 }
Statistics: Packets 1407, Bytes 695058
Next hop: 198.51.100.7 via ge-0/0/4.1
Label operation: Push 50008, Push 3460(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3460: None;
Label element ptr: 0xa192768
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 17, Stats IDs = { 536870930 }
Statistics: Packets 1418, Bytes 700492
Next hop: 198.51.100.8 via ge-0/0/4.2
Label operation: Push 50008, Push 3470(top)
```

```
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3470: None;
                Label element ptr: 0xa192790
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 18, Stats IDs = { 536870931 }
                Statistics: Packets 0, Bytes 0
                State: <Active Int NSR-incapable Programmed>
                Age: 12:46
                Validation State: unverified
                Announcement bits (3): 0-Resolve tree 6 1-Resolve tree 1 3-Resolve_IGP_FRR task
                AS path: I
                Session-IDs associated:
                Session-id: 370 Version: 0
                Thread: junos-main
```

show route extensive (BGP-SRTE routes)

```
user@host> show route extensive
inet.0: 22 destinations, 23 routes (21 active, 0 holddown, 1 hidden)
9.9.9.9-1 <c>/64 (1 entry, 0 announced):
        **SPRING-TE Preference: 8
                Next hop type: Indirect, Next hop index: 0
                Address: 0xdc33080
                Next-hop reference count: 1
                Next hop type: Router, Next hop index: 0
                Next hop: 1.2.2.2 via ge-0/0/2.0, selected
                Label element ptr: 0xdf671d0
                Label parent element ptr: 0x0
                Label element references: 11
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Protocol next hop: 299920
                Label operation: Push 800040
                Label TTL action: prop-ttl
                Load balance label: Label 800040: None;
                Composite next hop: 0xcd4f950 - INH Session ID: 0x0
```

```
                    Indirect next hop: 0xdc99a84 - INH Session ID: 0x0 Weight 0x1
                    State: <Active Int>
                    Local AS:   100
                    Age: 5d 17:37:19     Metric: 1     Metric2: 16777215
                    Validation State: unverified
                    Task: SPRING-TE
                    AS path:
                    SRTE Policy State:
                    SR Preference/Override: 200/100
                    Tunnel Source: Static configuration
                    Composite next hops: 1
                            Protocol next hop: 299920 Metric: 0
                            Label operation: Push 800040
                            Label TTL action: prop-ttl
                            Load balance label: Label 800040: None;
                            Composite next hop: 0xcd4f950 - INH Session ID: 0x0
                            Indirect next hop: 0xdc99a84 - INH Session ID: 0x0 Weight 0x1
                            Indirect path forwarding next hops: 1
                                    Next hop type: Router
                                    Next hop: 1.2.2.2 via ge-0/0/2.0
                                    Session Id: 0x0
                                    299920 /52 Originating RIB: mpls.0
                                      Metric: 0 Node path count: 1
                                      Forwarding nexthops: 1
                                            Next hop type: Router
                                            Next hop: 1.2.2.2 via ge-0/0/2.0
                                            Session Id: 0x141
```

Release Information

Command introduced before Junos OS Release 7.4.

`DeletePending` flag added to the command output in Junos OS Release 19.4R1.

show route forwarding-table

IN THIS SECTION

Syntax

```
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>
```

Syntax (MX Series Routers)

```
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>
```

Syntax (TX Matrix and TX Matrix Plus Routers)

```
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>
```

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: `bridge (ccc | destination | detail | extensive | interface-name | label | learning-vlan-id | matching | multicast | summary | table | vlan | vpn)`, `ethernet-switching`, `evpn`, `fibre-channel`, `fmembers`, `inet`, `inet6`, `iso`, `mcsnoop-inet`, `mcsnoop-inet6`, `mpls`, `satellite-inet`, `satellite-inet6`, `satellite-vpls`, `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*	(TX Matrix and TX matrix Plus routers only) (Optional) On a routing matrix composed of a TX Matrix router and T640 routers, 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 or T4000 routers, display information for the specified router (line-card chassis) connected to the TX Matrix Plus router. Replace *number* with the following values depending on the LCC configuration: • 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix. • 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix. • 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix. • 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
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	(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

Output Fields

Table 21 on page 469 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 21: 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` *`logical-system-name/routing-instance-name`* 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

Table 21: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Enabled protocols	The features and protocols that have been enabled for a given routing table. This field can contain the following values: • BUM hashing—BUM hashing is enabled. • MAC Stats—Mac Statistics is enabled. • Bridging—Routing instance is a normal layer 2 bridge. • No VLAN—No VLANs are associated with the bridge domain. • All VLANs—The `vlan-id all` statement has been enabled for this bridge domain. • Single VLAN—Single VLAN ID is associated with the bridge domain. • MAC action drop—New MACs will be dropped when the MAC address limit is reached. • Dual VLAN—Dual VLAN tags are associated with the bridge domain • No local switching—No local switching is enabled for this routing instance.. • Learning disabled—Layer 2 learning is disabled for this routing instance. • MAC limit reached—The maximum number of MAC addresses that was configured for this routing instance has been reached. • VPLS—The VPLS protocol is enabled. • No IRB l2-copy—The no-irb-layer-2-copy feature is enabled for this routing instance. • ACKed by all peers—All peers have acknowledged this routing instance. • BUM Pruning—BUM pruning is enabled on the VPLS instance. • Def BD VXLAN—VXLAN is enabled for the default bridge domain. • EVPN—EVPN protocol is enabled for this routing instance.	All levels

Table 21: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
	• Def BD OVSDB—Open vSwitch Database (OVSDB) is enabled on the default bridge domain. • Def BD Ingress replication—VXLAN ingress node replication is enabled on the default bridge domain. • L2 backhaul—Layer 2 backhaul is enabled. • FRR optimize—Fast reroute optimization • MAC pinning—MAC pinning is enabled for this bridge domain. • MAC Aging Timer—The MAC table aging time is set per routing instance. • EVPN VXLAN—This routing instance supports EVPN with VXLAN encapsulation. • PBBN—This routing instance is configured as a provider backbone bridged network. • PBN—This routing instance is configured as a provider bridge network. • ETREE—The ETREE protocol is enabled on this EVPN routing instance. • ARP/NDP suppression—EVPN ARP NDP suppression is enabled in this routing instance. • Def BD EVPN VXLAN—EVPN VXLAN is enabled for the default bridge domain. • MPLS control word—Control word is enabled for this MPLS routing instance.	
Address family	Address family (for example, `IP`, `IPv6`, `ISO`, `MPLS`, and `VPLS`).	All levels
Destination	Destination of the route.	`detail extensive`

Table 21: 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): • `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`

Table 21: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Flags	Route type flags: • `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. **NOTE**: For static routes that use point-to-point (P2P) outgoing interfaces, the next-hop address is not displayed in the output.	`detail extensive`

Table 21: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Next hop Type (Type)	Next-hop type. When the `detail` keyword is used, the next-hop type might be abbreviated (as indicated in parentheses): • `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 (mcrt)`—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 (recv)`—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. • `VxLAN Local`—EVPN Type 5 route in kernel.	`detail extensive`
Index	Software index of the next hop that is used to route the traffic for a given prefix.	`detail extensive` none

Table 21: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
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`
sftw	On Junos OS Evolved devices, `sftw` is another type of next-hop. A software next-hop can contain a link to another single next-hop, and hence it often works as an indirect next-hop. The usage of `sftw` helps to reduce churn in Junos OS Evolved objects by providing a common object where topology updates can be reflected. Further MPLS states can also be attached to these software next-hops. When a route is created, a `sftw` next-hop seems to be created whether it is the management port or the loopback interface.	`detail` `extensive` none

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
172.16.1.0/24         ifdn     0                    rslv   608     1 ge-2/0/1.0
172.16.1.0/32         iddn     0 172.16.1.0            recv   606     1 ge-2/0/1.0
172.16.1.1/32         user     0                    rjct    46     4
172.16.1.1/32         intf     0 172.16.1.1            locl   607     2
172.16.1.1/32         iddn     0 172.16.1.1            locl   607     2
172.16.1.255/32       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           locl   615     2
10.0.0.1/32        dest     0 10.0.0.1           locl   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           locl   611     2
10.1.1.1/32        iddn     0 10.1.1.1           locl   611     2
10.1.1.255/32      iddn     0 ff:ff:ff:ff:ff:ff  bcst   609     1 ge-2/0/1.0
10.206.0.0/16      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       locl   417     2
10.209.2.131/32    dest     0 10.209.2.131       locl   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                   locl    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           locl   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         locl   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          locl   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           locl   135     2
10.0.0.4/32        dest     0 10.0.0.4           locl   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 (EVPN Type 5 route with Type 2 and Type 5 route coexistence)

```
user@device> show route forwarding-table destination 10.1.1.20 table vrf1 extensive
Routing table: vrf1.inet [Index 9]
Internet:
Destination:  10.1.1.20/32
  Route type: user
  Route reference: 0                   Route interface-index: 0
  Multicast RPF nh index: 0
  P2mpidx: 0
  Flags: sent to PFE, VxLAN Local
  Nexthop:
```

```
Next-hop type: composite            Index: 2694     Reference: 7
Next-hop type: indirect             Index: 524326   Reference: 2
Next-hop type: unilist              Index: 524288   Reference: 5
Nexthop: 10.1.1.1
Next-hop type: unicast              Index: 1724     Reference: 15
Next-hop interface: xe-0/0/1.0    Weight: 0x0
Nexthop: 10.1.1.4  Next-hop type: unicast              Index: 1725     Reference: 15
Next-hop interface: xe-0/0/4.0    Weight: 0x0
```

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 192.0.2.2/30;
        }
    }
}
```

show route forwarding-table (Junos OS Evolved devices with next-hop type 'sftw')

```
user@host> show route forwarding-table | grep sftw
Internet:
Destination        Type RtRef Next hop           Type Index NhRef Netif
default            user     0                    sftw     7021     1 re0:mgmt-0.0
10.21.1.0/32       user     0                    sftw     7022     1 et-0/0/32.0

user@host> show route forwarding-table | grep sftw
Internet:
Destination        Type RtRef Next hop           Type Index NhRef Netif
default            user     0                    sftw     5021     1 re0:mgmt-0.0
10.22.1.0/32       user     0                    sftw     5022     1 et-0/0/32.0
```

Release Information

Command introduced before Junos OS Release 7.4.

Option `bridge-domain` introduced in Junos OS Release 7.5

Option `learning-vlan-id` introduced in Junos OS Release 8.4

Options `all` and `vlan` introduced in Junos OS Release 9.6.

RELATED DOCUMENTATION

show route instance

show route hidden

IN THIS SECTION

Syntax

```
show route hidden
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

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

Required Privilege Level

view

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.l3vpn.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 482.

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                       >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

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.l3vpn.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)
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

Understanding Hidden Routes

show route inactive-path

IN THIS SECTION

Syntax

```
show route inactive-path
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route inactive-path
<brief | detail | extensive | terse>
```

Description

Display routes for destinations that have no active route. An inactive route is a route that was not selected as the best path.

Options

none	Display all inactive routes.
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 | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 inactive-path

```
user@host> show route inactive-path

inet.0: 25 destinations, 26 routes (24 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both

10.12.100.12/30     [OSPF/10] 03:57:28, metric 1
                    > via so-0/3/0.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
10.0.0.0/8         [Direct/0] 04:39:56
                   > via fxp1.0

red.inet.0: 6 destinations, 8 routes (4 active, 0 holddown, 3 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both

10.12.80.0/30      [BGP/170] 04:38:17, localpref 100
                     AS path: 100 I
                   > to 10.12.80.1 via ge-6/3/2.0

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.l3vpn.0: 3 destinations, 3 routes (0 active, 0 holddown, 3 hidden)
Restart Complete

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 inactive-path detail

```
user@host> show route inactive-path detail

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

10.12.100.12/30 (2 entries, 1 announced)
         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:     1
                Age: 3:58:24    Metric: 1
```

```
                Area: 0.0.0.0
                Task: OSPF
                AS path: I

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

10.0.0.0/8 (2 entries, 0 announced)
         Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 1
                Next hop: via fxp1.0, selected
                State: <NotBest Int>
                Inactive reason: No difference
                Age: 4:40:52
                Task: IF
                AS path: I

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

10.12.80.0/30 (2 entries, 1 announced)
         BGP    Preference: 170/-101
                Next-hop reference count: 6
                Source: 10.12.80.1
                Next hop: 10.12.80.1 via ge-6/3/2.0, selected
                State: <Ext>
                Inactive reason: Route Preference
                Peer AS:   100
                Age: 4:39:13
                Task: BGP_100.10.12.80.1+179
                AS path: 100 I
                Localpref: 100
                Router ID: 10.0.0.0
```

show route inactive-path extensive

The output for the `show route inactive-path extensive` command is identical to that of the `show route inactive-path detail` command.

show route inactive-path terse

```
user@host> show route inactive-path 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
  10.12.100.12/30    O  10          1            >so-0/3/0.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
  10.0.0.0/8         D   0                       >fxp1.0

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
  10.12.80.0/30      B 170        100            >10.12.80.1      100 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.l3vpn.0: 3 destinations, 3 routes (0 active, 0 holddown, 3 hidden)
Restart Complete

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)
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route active-path

show route detail

show route extensive

show route terse

show route instance

IN THIS SECTION

Syntax

```
show route instance
<brief | detail | summary>
<instance-name>
```

```
<logical-system (all | logical-system-name)>
<operational>
```

Syntax (EX Series Switches and QFX Series)

```
show route instance
<brief | detail | summary>
<instance-name>
<operational>
```

Description

Display routing instance information.

Options

none	(Same as `brief`) Display standard information about all routing instances.
brief | detail | summary	(Optional) Display the specified level of output. If you do not specify a level of output, the system defaults to `brief`. (These options are not available with the `operational` keyword.)
instance-name	(Optional) Display information for all routing instances whose name begins with this string (for example, `cust1`, `cust11`, and `cust111` are all displayed when you run the `show route instance cust1` command).
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
operational	(Optional) Display operational routing instances.

Required Privilege Level

view

Output Fields

Table 22 on page 492 lists the output fields for the `show route instance` command. Output fields are listed in the approximate order in which they appear.

Table 22: show route instance Output Fields

Field Name	Field Description	Level of Output
Instance or *instance-name*	Name of the routing instance.	All levels
Operational Routing Instances	(`operational` keyword only) Names of all operational routing instances.	—
Type	Type of routing instance: `forwarding`, `l2vpn`, `no-forwarding`, `vpls`, `virtual-router`, or `vrf`.	All levels
State	State of the routing instance: `active` or `inactive`.	`brief` `detail` none
Interfaces	Name of interfaces belonging to this routing instance.	`brief` `detail` none
Restart State	Status of graceful restart for this instance: `Pending` or `Complete`.	`detail`
Path selection timeout	Maximum amount of time, in seconds, remaining until graceful restart is declared complete. The default is `300`.	`detail`
Tables	Tables (and number of routes) associated with this routing instance.	`brief` `detail` none
Route-distinguisher	Unique route distinguisher associated with this routing instance.	`detail`

Table 22: show route instance Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Vrf-import	VPN routing and forwarding instance import policy name.	`detail`
Vrf-export	VPN routing and forwarding instance export policy name.	`detail`
Vrf-import-target	VPN routing and forwarding instance import target community name.	`detail`
Vrf-export-target	VPN routing and forwarding instance export target community name.	`detail`
Vrf-edge-protection-id	Context identifier configured for edge-protection.	`detail`
Fast-reroute-priority	Fast reroute priority setting for a VPLS routing instance: `high`, `medium`, or `low`. The default is `low`.	`detail`
Restart State	Restart state: • `Pending:`*`protocol-name`*—List of protocols that have not yet completed graceful restart for this routing table. • `Complete`—All protocols have restarted for this routing table.	`detail`
Primary rib	Primary table for this routing instance.	`brief` none `summary`
Active/holddown/ hidden	Number of active, hold-down, and hidden routes.	All levels

Sample Output

show route instance

```
user@host> show route instance
Instance             Type
         Primary RIB                                     Active/holddown/hidden
master               forwarding
         inet.0                                          16/0/1
         iso.0                                           1/0/0
         mpls.0                                          0/0/0
         inet6.0                                         2/0/0
         l2circuit.0                                     0/0/0
__juniper_private1__ forwarding
         __juniper_private1__.inet.0                     12/0/0
         __juniper_private1__.inet6.0                    1/0/0
```

show route instance detail (VPLS Routing Instance)

```
user@host> show route instance detail test-vpls
test-vpls:
  Router ID: 0.0.0.0
  Type: vpls              State: Active
  Interfaces:
    lsi.1048833
    lsi.1048832
    fe-0/1/0.513
  Route-distinguisher: 10.255.37.65:1
  Vrf-import: [ __vrf-import-test-vpls-internal__ ]
  Vrf-export: [ __vrf-export-test-vpls-internal__ ]
  Vrf-import-target: [ target:300:1 ]
  Vrf-export-target: [ target:300:1 ]
  Vrf-edge-protection-id:  166.1.3.1  Fast-reroute-priority: high
  Tables:
    test-vpls.l2vpn.0          : 3 routes (3 active, 0 holddown, 0 hidden)
```

show route instance operational

```
user@host> show route instance operational
Operational Routing Instances:

master
default
```

show route instance summary

```
user@host> show route instance summary
Instance           Type           Primary rib            Active/holddown/hidden
master             forwarding
                                  inet.0                 15/0/1
                                  iso.0                  1/0/0
                                  mpls.0                 35/0/0
                                  l3vpn.0                0/0/0
                                  inet6.0                2/0/0
                                  l2vpn.0                0/0/0
                                  l2circuit.0            0/0/0
BGP-INET           vrf
                                  BGP-INET.inet.0        5/0/0
                                  BGP-INET.iso.0         0/0/0
                                  BGP-INET.inet6.0       0/0/0
BGP-L              vrf
                                  BGP-L.inet.0           5/0/0
                                  BGP-L.iso.0            0/0/0
                                  BGP-L.mpls.0           4/0/0
                                  BGP-L.inet6.0          0/0/0
L2VPN              l2vpn
                                  L2VPN.inet.0           0/0/0
                                  L2VPN.iso.0            0/0/0
                                  L2VPN.inet6.0          0/0/0
                                  L2VPN.l2vpn.0          2/0/0
LDP                vrf
                                  LDP.inet.0             4/0/0
                                  LDP.iso.0              0/0/0
                                  LDP.mpls.0             0/0/0
                                  LDP.inet6.0            0/0/0
                                  LDP.l2circuit.0        0/0/0
```

```
OSPF                vrf
                              OSPF.inet.0             7/0/0
                              OSPF.iso.0              0/0/0
                              OSPF.inet6.0            0/0/0
RIP                 vrf
                              RIP.inet.0              6/0/0
                              RIP.iso.0               0/0/0
                              RIP.inet6.0             0/0/0
STATIC              vrf
                              STATIC.inet.0           4/0/0
                              STATIC.iso.0            0/0/0
                              STATIC.inet6.0          0/0/0
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Transporting IPv6 Traffic Across IPv4 Using Filter-Based Tunneling

Example: Configuring the Helper Capability Mode for OSPFv3 Graceful Restart

show route next-hop

IN THIS SECTION

Syntax

```
show route next-hop next-hop
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route next-hop next-hop
<brief | detail | extensive | terse>
```

Description

Display the entries in the routing table that are being sent to the specified next-hop address.

Options

brief | detail | extensive | terse	(Optional) Display the specified level of ouput.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
next-hop	Next-hop address.

Required Privilege Level

view

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 next-hop

```
user@host> show route next-hop 192.168.71.254

inet.0: 18 destinations, 18 routes (17 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both

10.10.0.0/16       *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
10.209.0.0/16      *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
172.16.0.0/12      *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
192.168.0.0/16     *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
192.168.102.0/23   *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
207.17.136.0/24    *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
207.17.136.192/32  *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

red.inet.0: 4 destinations, 5 routes (4 active, 0 holddown, 0 hidden)
```

```
Restart Complete

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

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 next-hop terse

```
user@host> show route next-hop 192.168.71.254 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
* 10.10.0.0/16       S   5                       >192.168.71.254
* 10.209.0.0/16      S   5                       >192.168.71.254
* 172.16.0.0/12      S   5                       >192.168.71.254
* 192.168.0.0/16     S   5                       >192.168.71.254
* 192.168.102.0/23   S   5                       >192.168.71.254
* 207.17.136.0/24    S   5                       >192.168.71.254
* 207.17.136.192/32  S   5                       >192.168.71.254

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

red.inet.0: 4 destinations, 5 routes (4 active, 0 holddown, 0 hidden)
Restart Complete

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
```

```
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)
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route output

IN THIS SECTION

Syntax

```
show route output (address ip-address | interface interface-name)
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route output (address ip-address | interface interface-name)
<brief | detail | extensive | terse>
```

Description

Display the entries in the routing table learned through static routes and interior gateway protocols that are to be sent out the interface with either the specified IP address or specified name.

To view routes advertised to a neighbor or received from a neighbor for the BGP protocol, use the `show route advertising-protocol bgp` and `show route receive-protocol bgp` commands instead.

Options

address *ip-address*	Display entries in the routing table that are to be sent out the interface with the specified IP address.
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`.
interface *interface-name*	Display entries in the routing table that are to be sent out the interface with the specified name.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 output address

```
user@host> show route output address 172.16.36.1/24

inet.0: 28 destinations, 30 routes (27 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.36.0/24       *[Direct/0] 00:19:56
                    > via so-0/1/2.0
                    [OSPF/10] 00:19:55, metric 1
                    > via so-0/1/2.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

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

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

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

private1__.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

show route output address detail

```
user@host> show route output address 172.16.36.1 detail

inet.0: 28 destinations, 30 routes (27 active, 0 holddown, 1 hidden)
172.16.36.0/24 (2 entries, 0 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 1
                Next hop: via so-0/1/2.0, selected
                State: <Active Int>
                Age: 23:00
                Task: IF
                AS path: I
         OSPF   Preference: 10
                Next-hop reference count: 1
                Next hop: via so-0/1/2.0, selected
                State: <Int>
                Inactive reason: Route Preference
                Age: 22:59      Metric: 1
                Area: 0.0.0.0
                Task: OSPF
                AS path: I

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

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

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

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

private1__.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

show route output address extensive

The output for the `show route output address extensive` command is identical to that of the `show route output address detail` command. For sample output, see "show route output address detail" on page 503.

show route output address terse

```
user@host> show route output address 172.16.36.1 terse

inet.0: 28 destinations, 30 routes (27 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 172.16.36.0/24        D   0                       >so-0/1/2.0
                     O  10          1            >so-0/1/2.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

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

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

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

private1__.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route protocol

IN THIS SECTION

Syntax

```
show route protocol protocol
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route protocol protocol
<brief | detail | extensive | terse>
```

Syntax (PTX and ACX Series Switches)

```
show route protocol protocol
(index index | prefix prefix)
<brief | detail | extensive | terse>
```

Operational mode

`cli` mode

`cli-pfe` mode.

Description

Display the route entries in the routing table that were learned from a particular protocol. Displays details about routes installed in `cli-pfe` mode after FIB compression of FIB compression statistics (PTX and ACX series devices only).

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 | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
protocol	Protocol from which the route was learned:

- `access`—Access route for use by DHCP application
- `access-internal`—Access-internal route for use by DHCP application
- `aggregate`—Locally generated aggregate route
- `arp`—Route learned through the Address Resolution Protocol

- `atmvpn`—Asynchronous Transfer Mode virtual private network
- `bgp`—Border Gateway Protocol
- `ccc`—Circuit cross-connect
- `direct`—Directly connected route
- `dvmrp`—Distance Vector Multicast Routing Protocol
- `esis`—End System-to-Intermediate System
- `flow`—Locally defined flow-specification route
- `frr`—Precomputed protection route or backup route used when a link goes down
- `isis`—Intermediate System-to-Intermediate System
- `ldp`—Label Distribution Protocol
- `l2circuit`—Layer 2 circuit
- `l2vpn`—Layer 2 virtual private network
- `local`—Local address
- `mpls`—Multiprotocol Label Switching
- `msdp`—Multicast Source Discovery Protocol
- `ospf`—Open Shortest Path First versions 2 and 3
- `ospf2`—Open Shortest Path First versions 2 only
- `ospf3`—Open Shortest Path First version 3 only
- `pim`—Protocol Independent Multicast
- `rip`—Routing Information Protocol
- `ripng`—Routing Information Protocol next generation
- `rsvp`—Resource Reservation Protocol
- `rtarget`—Local route target virtual private network
- `static`—Statically defined route

- `tunnel`—Dynamic tunnel
- `vpn`—Virtual private network

NOTE: EX Series switches run a subset of these protocols. See the switch CLI for details.

Required Privilege Level

view

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 protocol access

```
user@host> show route protocol access
inet.0: 30380 destinations, 30382 routes (30379 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

13.160.0.3/32      *[Access/13] 00:00:09
                    > to 13.160.0.2 via fe-0/0/0.0
13.160.0.4/32      *[Access/13] 00:00:09
                    > to 13.160.0.2 via fe-0/0/0.0
13.160.0.5/32      *[Access/13] 00:00:09
                    > to 13.160.0.2 via fe-0/0/0.0
```

show route protocol arp

```
user@host>  show route protocol arp
inet.0: 43 destinations, 43 routes (42 active, 0 holddown, 1 hidden)

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

cust1.inet.0: 1033 destinations, 2043 routes (1033 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

20.20.1.3/32       [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.4/32       [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.5/32       [ARP/4294967293] 00:04:32, from 20.20.1.1
                      Unusable
20.20.1.6/32       [ARP/4294967293] 00:04:34, from 20.20.1.1
                      Unusable
20.20.1.7/32       [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.8/32       [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.9/32       [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.10/32      [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.11/32      [ARP/4294967293] 00:04:33, from 20.20.1.1
                      Unusable
20.20.1.12/32      [ARP/4294967293] 00:04:33, from 20.20.1.1
                      Unusable
20.20.1.13/32      [ARP/4294967293] 00:04:33, from 20.20.1.1
                      Unusable
...
```

show route protocol bgp

```
user@host> show route protocol bgp 192.168.64.0/21
inet.0: 335832 destinations, 335833 routes (335383 active, 0 holddown, 450 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
192.168.64.0/21       *[BGP/170] 6d 10:41:16, localpref 100, from 192.168.69.71
                        AS path: 10458 14203 2914 4788 4788 I
                      > to 192.168.167.254 via fxp0.0
```

show route protocol direct

```
user@host> show route protocol direct

inet.0: 335843 destinations, 335844 routes (335394 active, 0 holddown, 450 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.8.0/24         *[Direct/0] 17w0d 10:31:49
                      > via fe-1/3/1.0
10.255.165.1/32    *[Direct/0] 25w4d 04:13:18
                      > via lo0.0
172.16.30.0/24      *[Direct/0] 17w0d 23:06:26
                      > via fe-1/3/2.0
192.168.164.0/22   *[Direct/0] 25w4d 04:13:20
                      > via fxp0.0

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

47.0005.80ff.f800.0000.0108.0001.0102.5516.5001/152
                   *[Direct/0] 25w4d 04:13:21
                      > via lo0.0

inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::10:255:165:1/128
                   *[Direct/0] 25w4d 04:13:21
                      > via lo0.0
fe80::2a0:a5ff:fe12:ad7/128
                   *[Direct/0] 25w4d 04:13:21
                      > via lo0.0
```

show route protocol frr

```
user@host> show route protocol frr
inet.0: 43 destinations, 43 routes (42 active, 0 holddown, 1 hidden)

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

cust1.inet.0: 1033 destinations, 2043 routes (1033 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

20.20.1.3/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.3 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.4/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.4 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.5/32       *[FRR/200] 00:05:35, from 20.20.1.1
                    > to 20.20.1.5 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.6/32       *[FRR/200] 00:05:37, from 20.20.1.1
                    > to 20.20.1.6 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.7/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.7 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.8/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.8 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.9/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.9 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.10/32      *[FRR/200] 00:05:38, from 20.20.1.1
...
```

show route protocol ldp

```
user@host> show route protocol ldp
inet.0: 12 destinations, 13 routes (12 active, 0 holddown, 0 hidden)

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

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

192.168.16.1/32    *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0, Push 100000
192.168.17.1/32    *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0

private1__.inet.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

mpls.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

100064             *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0, Pop
100064(S=0)        *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0, Pop
100080             *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0, Swap 100000
```

show route protocol ospf (Layer 3 VPN)

```
user@host> show route protocol ospf
inet.0: 40 destinations, 40 routes (39 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

10.39.1.4/30       *[OSPF/10] 00:05:18, metric 4
                    > via t3-3/2/0.0
10.39.1.8/30        [OSPF/10] 00:05:18, metric 2
                    > via t3-3/2/0.0
10.255.14.171/32   *[OSPF/10] 00:05:18, metric 4
                    > via t3-3/2/0.0
10.255.14.179/32   *[OSPF/10] 00:05:18, metric 2
                    > via t3-3/2/0.0
172.16.233.5/32       *[OSPF/10] 20:25:55, metric 1

VPN-AB.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.39.1.16/30       [OSPF/10] 00:05:43, metric 1
                    > via so-0/2/2.0
10.255.14.173/32   *[OSPF/10] 00:05:43, metric 1
```

```
                    > via so-0/2/2.0
172.16.233.5/32       *[OSPF/10] 20:26:20, metric 1
```

show route protocol rip

```
user@host> show route protocol rip
inet.0: 26 destinations, 27 routes (25 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

VPN-AB.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
10.255.14.177/32   *[RIP/100] 20:24:34, metric 2
                    > to 10.39.1.22 via t3-0/2/2.0
172.16.233.9/32       *[RIP/100] 00:03:59, metric 1
```

show route protocol rsvp extensive

```
user@host> show route protocol rsvp extensive
inet.0: 37 destinations, 37 routes (36 active, 0 holddown, 1 hidden)

inet.3: 3 destinations, 4 routes (3 active, 0 holddown, 0 hidden)
192.168.4.4/32 (2 entries, 1 announced)
        State: <FlashAll>
        *RSVP   Preference: 7/1
                Next hop type: Router, Next hop index: 0
                Address: 0xaad4fbc
                Next-hop reference count: 2, key opaque handle: 0x0, non-key opaque handle: 0x0
                Next hop: 192.168.0.1 via ge-0/0/0.0 weight 0x1, selected
                Label-switched-path PE1_to_PE2_1
                Label operation: Push 299920
                Label TTL action: prop-ttl
                Load balance label: Label 299920: None;
                Label element ptr: 0x9938be8
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 1
                Label element lsp id: 2
                Session Id: 0
                Next hop: 192.168.1.1 via ge-0/0/1.0 weight 0x8001 uflags Mark connection
protection tlv
```

```
                  Label-switched-path Bypass->192.168.0.1
                  Label operation: Push 299920
                  Label TTL action: prop-ttl
                  Load balance label: Label 299920: None;
                  Label element ptr: 0x9938eb8
                  Label parent element ptr: 0x9938be8
                  Label element references: 2
                  Label element child references: 0
                  Label element lsp id: 2
                  Session Id: 0
                  State: <Active Int>
                  Local AS:   100
                  Age: 1:20       Metric: 100
                  Validation State: unverified
                  Actual IGP metric: 2
                  Task: RSVP
                  Announcement bits (2): 0-Resolve tree 3 3-Resolve tree 1
                  AS path: I
                  Thread: junos-main
```

show route protocol index prefix detail (ACX7100-32C, ACX7100-48L, and ACX7509 Devices)

Displays if a route is installed or not when FIB compression is enabledIf the route is not installed, the command displays the details of the parent route installed, that compressed the route.

```
user@root:pfe> show route proto 2 index 0 prefix 200.1.0.16 detail
Protocol: 2
Protocol     : IPv4
Table        : default
Prefix       : 200.1.0.16 (primary)
NH           : 25035 (software)
Flags        : 0x00008000
Details      :
guid          : 889058232562
type          :  user
nhid          :  25035
Forwarding state:
installed?    :  no
(Installed parent: 200.1.0/23)
```

Release Information

Command introduced before Junos OS Release 7.4.

`ospf2` and `ospf3` options introduced in Junos OS Release 9.2.

`ospf2` and `ospf3` options introduced in Junos OS Release 9.2 for EX Series switches.

`flow` option introduced in Junos OS Release 10.0.

`flow` option introduced in Junos OS Release 10.0 for EX Series switches.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route receive-protocol

IN THIS SECTION

Syntax

```
show route receive-protocol protocol neighbor-address
 <brief | detail | extensive | terse>
<logical-system (all | logical-system-name)
```

Syntax (EX Series Switches)

```
show route receive-protocol protocol neighbor-address
<brief | detail | extensive | terse>
```

Description

Display the routing information as it was received through a particular neighbor using a particular dynamic routing protocol.

Options

brief | detail | extensive | terse	(Optional) Display the specified level of output.
logical-system (all | ***logical-system-name*****)**	(Optional) Perform this operation on all logical systems or on a particular logical system.
protocol neighbor-address	Protocol transmitting the route (`bgp`, `dvmrp`, `msdp`, `pim`, `rip`, or `ripng`) and address of the neighboring router from which the route entry was received.

Additional Information

The output displays the selected routes and the attributes with which they were received, but does not show the effects of import policy on the routing attributes.

Required Privilege Level

view

Output Fields

Table 23 on page 517 describes the output fields for the `show route receive-protocol` command. Output fields are listed in the approximate order in which they appear.

Table 23: show route receive-protocol Output Fields

Field Name	Field Description	Level of Output
routing-table-name	Name of the routing table—for example, inet.0.	All levels
number destinations	Number of destinations for which there are routes in the routing table.	All levels
number routes	Number of routes in the routing table and total number of routes in the following states: • `active` • `holddown` (routes that are in pending state before being declared inactive) • `hidden` (routes that are not used because of a routing policy)	All levels
Prefix	Destination prefix.	none `brief`
MED	Multiple exit discriminator value included in the route.	none `brief`

Table 23: show route receive-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
destination-prefix (entry, announced)	Destination prefix. 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.	`detail extensive`
Accepted LongLivedStale	The LongLivedStale flag indicates that the route was marked LLGR-stale by this router, as part of the operation of LLGR receiver mode. Either this flag or the LongLivedStaleImport flag may be displayed for a route. Neither of these flags are displayed at the same time as the Stale (ordinary GR stale) flag.	`detail extensive`
Accepted LongLivedStaleImport	The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy. Either this flag or the LongLivedStale flag may be displayed for a route. Neither of these flags are displayed at the same time as the Stale (ordinary GR stale) flag. Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and import into the inet.0 routing table	`detail extensive`
ImportAccepted LongLivedStaleImport	Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and imported into the inet.0 routing table The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy.	`detail extensive`
Route Distinguisher	64-bit prefix added to IP subnets to make them unique.	`detail extensive`
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.	`detail extensive`
VPN Label	Virtual private network (VPN) label. Packets are sent between CE and PE routing devices by advertising VPN labels. VPN labels transit over either an RSVP or an LDP label-switched path (LSP) tunnel.	`detail extensive`

Table 23: show route receive-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Next hop	Next hop to the destination. An angle bracket (>) indicates that the route is the selected route.	All levels
Localpref or Lclpref	Local preference value included in the route.	All levels
AS path	Autonomous system (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: • `I`—IGP. • `E`—EGP. • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—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 the AS-path merge process, as defined in RFC 4893. • `[ ]`—If more than one AS number is configured on the router, 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. **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.	All levels

Table 23: show route receive-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Route Labels	Stack of labels carried in the BGP route update.	`detail extensive`
Cluster list	(For route reflected output only) Cluster ID sent by the route reflector.	`detail extensive`
Originator ID	(For route reflected output only) Address of routing device that originally sent the route to the route reflector.	`detail extensive`
Communities	Community path attribute for the route.	`detail extensive`
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.	`detail extensive`
Attrset AS	Number, local preference, and path of the AS that originated the route. These values are stored in the `Attrset` attribute at the originating routing device.	`detail extensive`
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).	`detail extensive`
control flags	Control flags: `none` or `Site Down`.	`detail extensive`
mtu	Maximum transmission unit (MTU) of the Layer 2 circuit.	`detail extensive`

Sample Output

show route receive-protocol bgp

```
user@host> show route receive-protocol bgp 10.255.245.215

inet.0: 28 destinations, 33 routes (27 active, 0 holddown, 1 hidden)
Prefix                  Next hop              MED     Lclpref    AS path
```

```
10.22.1.0/24            10.255.245.215       0       100        I
10.22.2.0/24            10.255.245.215       0       100        I
```

show route receive-protocol bgp detail (BGP Multicast)

```
user@host> show route receive-protocol bgp 10.4.6.1 detail
bgpmcast.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
* 4:3:0:0:32:10.2.20.20:32:225.1.1.1:10.4.4.4:10.4.0.1/240 (1 entry, 1 announced)
     Accepted
     Nexthop: 10.4.0.1
     Localpref: 100
     AS path: 65002 I
     Communities: target:10.4.4.4:0
     Tunnel type: AnyEncap, RPF tunnel:, Remote end point: 10.1.4.4
     Tunnel type: AnyEncap, Remote end point: 10.2.4.4
     Tunnel type: AnyEncap, Remote end point: 10.3.4.4
     Tunnel type: AnyEncap, Remote end point: 10.4.6.4
     Tunnel type: AnyEncap, Remote end point: 10.4.20.4
```

Show route receive protocol (Segment Routing Traffic Engineering)

```
show route receive protocol bgp 10.1.1.4
bgp.inetcolor.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

* 50-10.4.4.4-1234<sr6>/96 (1 entry, 0 announced)
     Import Accepted
     Distinguisher: 50
     Color: 1234
     Nexthop: 10.1.1.4
     Localpref: 100
     AS path: 65003 I
     Communities: target:10.1.1.1:1


inetcolor.0: 6 destinations, 7 routes (6 active, 0 holddown, 0 hidden)
* 10.4.4.4-1234<c6>/64 (1 entry, 1 announced)
     Import Accepted
     Color: 1234
     Nexthop: 10.1.1.4
     Localpref: 100
```

```
    AS path: 65003 I
    Communities: target:10.1.1.1:1


user@host# run show route receive-protocol bgp 2001:db8:5001:1::4
bgp.inet6color.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

* 50-2001:db8:1::4-1234<sr6>/192 (1 entry, 0 announced)
    Import Accepted
    Distinguisher: 50
    Color: 1234
    Nexthop: ::ffff:10.1.1.4
    Localpref: 100
    AS path: 65003 I
    Communities: target:10.1.1.1:1


inet6color.0: 6 destinations, 7 routes (6 active, 0 holddown, 0 hidden)
* 2001:db8::5-1234<c6>/160 (1 entry, 1 announced)
    Import Accepted
    Color: 1234
    Nexthop: ::ffff:10.1.1.5
    Localpref: 100
    AS path: 65003 I
    Communities: target:10.1.1.1:1
```

Release Information

Command introduced before Junos OS Release 7.4.

show route table

IN THIS SECTION

Syntax

```
show route table routing-table-name
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches, QFX Series Switches)

```
show route table routing-table-name
<brief | detail | extensive | terse>
```

Description

Display the route entries in a particular routing table.

Options

brief | detail | extensive | terse	(Optional) Display the specified level of output.

logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system. This option is only supported on Junos OS.
routing-table-name	Display route entries for all routing tables whose names begin with this string (for example, inet.0 and inet6.0 are both displayed when you run the `show route table inet` command).

Required Privilege Level

view

Output Fields

Table 24 on page 524 describes the output fields for the `show route table` command. Output fields are listed in the approximate order in which they appear.

Table 24: show route table Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).

Table 24: show route table Output Fields *(Continued)*

Field Name	Field Description

Restart complete	All protocols have restarted for this routing table.
	
	Restart state:
	
	• `Pending:`*`protocol-name`*—List of protocols that have not yet completed graceful restart for this routing table.
	
	• `Complete`—All protocols have restarted for this routing table.
	
	For example, if the output shows-
	
	• `LDP.inet.0            : 5 routes (4 active, 1 holddown, 0 hidden)`
	`    Restart Pending: OSPF LDP VPN`
	
	This indicates that `OSPF`, `LDP`, and `VPN` protocols did not restart for the `LDP.inet.0` routing table.
	
	• `vpls_1.l2vpn.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)`
	`Restart Complete`
	
	This indicates that all protocols have restarted for the `vpls_1.l2vpn.0` routing table.

number destinations	Number of destinations for which there are routes in the routing table.
number routes	Number of routes in the routing table and total number of routes in the following states: • `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 24: show route table Output Fields *(Continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th></tr>
<tr><td>route-destination (entry, announced)</td><td>Route destination (for example:10.0.0.1/24). The <code>entry</code> value is the number of routes for this destination, and the <code>announced</code> value is the number of routes being announced for this destination. Sometimes the route destination is presented in another format, such as:

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).

neighbor-address—Address of the neighbor.
control-word-status—Whether the use of the control word has been negotiated for this virtual circuit: <code>NoCtrlWord</code> or <code>CtrlWord</code>.
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.
<code>vc-id</code>—Virtual circuit identifier.
source—Source of the advertisement: <code>Local</code> or <code>Remote</code>.

<code>inclusive multicast Ethernet tag route</code>—Type of route destination represented by (for example, 3:100.100.100.10:100::0::10::100.100.100.10/384):

route distinguisher—(8 octets) Route distinguisher (RD) must be the RD of the EVPN instance (EVI) that is advertising the NLRI.
Ethernet tag ID—(4 octets) Identifier of the Ethernet tag. Can set to 0 or to a valid Ethernet tag value.
IP address length—(1 octet) Length of IP address in bits.
originating router’s IP address—(4 or 16 octets) Must set to the provider edge (PE) device’s IP address. This address should be common for all EVIs on the PE device, and may be the PE device's loopback address.

</td></tr>
</table>

Table 24: show route table Output Fields *(Continued)*

Field Name	Field Description
label stacking	(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. • `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).
[*protocol*, *preference*]	Protocol from which the route was learned and the preference value for the route. • `+`—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. 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.
Level	(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.
Route Distinguisher	IP subnet augmented with a 64-bit prefix.
PMSI	Provider multicast service interface (MVPN routing table).

Table 24: show route table Output Fields *(Continued)*

Field Name	Field Description
Next-hop type	Type of next hop. For a description of possible values for this field, see Table 25 on page 533.
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	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: • 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).

Table 24: show route table Output Fields *(Continued)*

Field Name	Field Description
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 26 on page 535.
Local AS	AS number of the local routing devices.
Age	How long the route has been known.
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.
Metric*n*	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.
MED-plus-IGP	Metric value for BGP path selection to which the IGP cost to the next-hop destination has been added.
TTL-Action	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.
Task	Name of the protocol that has added the route.

Table 24: show route table Output Fields *(Continued)*

Field Name	Field Description
Announcement bits	The number of BGP peers or protocols to which Junos OS has announced this route, followed by the list of the recipients of the announcement. Junos OS can also announce the route to the kernel routing table (KRT) for installing the route into the Packet Forwarding Engine, to a resolve tree, a Layer 2 VC, or even a VPN. For example, *n*-`Resolve inet` indicates that the specified route is used for route resolution for next hops found in the routing table. • *n*—An index used by Juniper Networks customer support only.
AS path	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: • `I`—IGP. • `E`—EGP. • `Recorded`—The AS path is recorded by the sample process (sampled). • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—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. **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.

Table 24: show route table Output Fields *(Continued)*

Field Name	Field Description
validation-state	(BGP-learned routes) Validation status of the route: • `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. • `Unverified`—Indicates that the origin of the prefix is not verified against the database. This is because the database got populated and the validation is not called for in the BGP import policy, although origin validation is enabled, or the origin validation is not enabled for the BGP peers. • `Valid`—Indicates that the prefix and autonomous system pair are found in the database.
FECs bound to route	Indicates point-to-multipoint root address, multicast source address, and multicast group address when multipoint LDP (M-LDP) inband signaling is configured.
Primary Upstream	When multipoint LDP with multicast-only fast reroute (MoFRR) is configured, indicates the primary upstream path. MoFRR transmits a multicast join message from a receiver toward a source on a primary path, while also transmitting a secondary multicast join message from the receiver toward the source on a backup path.
RPF Nexthops	When multipoint LDP with MoFRR is configured, indicates the reverse-path forwarding (RPF) next-hop information. Data packets are received from both the primary path and the secondary paths. The redundant packets are discarded at topology merge points due to the RPF checks.
Label	Multiple MPLS labels are used to control MoFRR stream selection. Each label represents a separate route, but each references the same interface list check. Only the primary label is forwarded while all others are dropped. Multiple interfaces can receive packets using the same label.
weight	Value used to distinguish MoFRR primary and backup routes. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible.

Table 24: show route table Output Fields *(Continued)*

Field Name	Field Description
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 27 on page 539 for all possible values for this field.
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).
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 LongLivedStale	The LongLivedStale flag indicates that the route was marked LLGR-stale by this router, as part of the operation of LLGR receiver mode. Either this flag or the LongLivedStaleImport flag might be displayed for a route. Neither of these flags is displayed at the same time as the Stale (ordinary GR stale) flag.

Table 24: show route table Output Fields *(Continued)*

Field Name	Field Description
Accepted LongLivedStaleImpor t	The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy. Either this flag or the LongLivedStale flag might be displayed for a route. Neither of these flags is displayed at the same time as the Stale (ordinary GR stale) flag. Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and import into the inet.0 routing table
ImportAccepted LongLivedStaleImpor t	Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and imported into the inet.0 routing table The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy.
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 25 on page 533 describes all possible values for the Next-hop Types output field.

Table 25: Next-hop Types Output Field Values

Next-Hop Type	Description
Broadcast (bcast)	Broadcast next hop.

Table 25: Next-hop Types Output Field Values *(Continued)*

Next-Hop Type	Description
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.
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.

Table 25: Next-hop Types Output Field Values *(Continued)*

Next-Hop Type	Description
Reject (rjct)	Discard. An ICMP unreachable message was sent.
Resolve (rslv)	Resolving next hop.
Routed multicast (mcrt)	Regular multicast next hop.
Router	A specific node or set of nodes to which the routing device forwards packets that match the route prefix. To qualify as a next-hop type router, the route must meet the following criteria: • 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 26 on page 535 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 26: State Output Field Values

Value	Description
Accounting	Route needs accounting.
Active	Route is active.

Table 26: State Output Field Values *(Continued)*

Value	Description
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.

Table 26: State Output Field Values *(Continued)*

Value	Description
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).
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.

Table 26: State Output Field Values *(Continued)*

Value	Description
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: • 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.
VxlanLocalRT	Route is an EVPN Type 5 route (IP prefix route).

Table 27 on page 539 describes the possible values for the Communities output field.

Table 27: Communities Output Field Values

Value	Description
area-number	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.
bandwidth: *local AS number:link-bandwidth-number*	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.
domain-id	Unique configurable number that identifies the OSPF domain.
GBP Tag	Assigned Group Based Policy (GBP) tag from 1 through 65535.
domain-id-vendor	Unique configurable number that further identifies the OSPF domain.
link-bandwidth-number	Link-bandwidth number: from 0 through 4,294,967,295 (bytes per second).
local AS number	Local AS number: from 1 through 65,535.
options	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.
origin	(Used with VPNs) Identifies where the route came from.
ospf-route-type	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 be0); 7 for NSSA routes; or 129 for sham link endpoint addresses.
route-type-vendor	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*.

Table 27: Communities Output Field Values *(Continued)*

Value	Description
rte-type	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*.
target	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.
unknown IANA	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.
unknown OSPF vendor community	Incoming IANA codes with a value above 0x8000. This code of the BGP extended community attribute is accepted, but it is not recognized.
evpn-mcast-flags	Identifies the value in the multicast flags extended community, which includes bits that indicate whether snooping or optimized intersubnet multicast (OISM) is enabled. For example: • 0x01—IGMP snooping bit • 0x02—MLD snooping bit • 0x08—OISM bit
evpn-l2-info	Identifies whether Multihomed Proxy MAC and IP Address Route Advertisement is enabled. A value of 0x20 indicates that the proxy bit is set. . Use the `show bridge mac-ip-table extensive` statement to determine whether the MAC and IP address route was learned locally or from a PE device.

Sample Output

show route table bgp.l2vpn.0

```
user@host> show route table bgp.l2vpn.0
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 evpn-vs2.evpn.0 extensive

```
user@host> show route table evpn-vs2.evpn.0 extensive
evpn-vs2.evpn.0: 30 destinations, 30 routes (30 active, 0 holddown, 0 hidden)
Restart Complete
2:10.3.3.3:2::601: :2c:6b:f5:f2:87:f0/304 MAC/IP (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 10.3.3.3:2
                Next hop type: Indirect, Next hop index: 0
                Address: 0x76b66dc
                Next-hop reference count: 32
                Source: 10.2.2.2
                Protocol next hop: 10.3.3.3
                Indirect next hop: 0x2 no-forward INH Session ID: 0
                State: <Secondary Active Ext>
                Local AS:   102 Peer AS:   201
                Age: 22:56:12     Metric2: 1
                Validation State: unverified
                Task: BGP_201.10.2.2.2
                Announcement bits (1): 0-evpn-vs2-evpn
                AS path: 201 203 I
                Communities: target:5:5 encapsulation:vxlan(0x8) evpn-default-gateway gbp-tag:
200
```

show route table bgpmcast.0 extensive

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


4:3:0:0:32:10.2.20.20:32:225.1.1.1:10.4.4.4:10.20.20.20/240 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Tunnel type: AnyEncap, RPF tunnel:, Remote end point: 10.1.4.4
                Tunnel type: AnyEncap, Remote end point: 10.2.4.4
                Tunnel type: AnyEncap, Remote end point: 10.3.4.4
                Tunnel type: AnyEncap, Remote end point: 10.4.6.4
                Next hop type: Indirect, Next hop index: 0
                Address: 0xc54639c
                Next-hop reference count: 1
                Source: 10.1.1.1
                Protocol next hop: 10.20.20.20
                Indirect next hop: 0x2 no-forward INH Session ID: 0x0
                State: <Active Int Ext>
                Local AS:   65100 Peer AS:   65100
                Age: 30         Metric2: 0
                Validation State: unverified
                Task: BGP_100.1.1.1.1
                Announcement bits (1): 0-bgpmcast global task
                AS path: I
                Communities: target:10.4.4.4:0
                Accepted
                Localpref: 100
                Router ID: 10.1.1.1
                Indirect next hops: 1
                        Protocol next hop: 10.20.20.20
                        Indirect next hop: 0x2 no-forward INH Session ID: 0x0
                        Indirect path forwarding next hops: 1
                                Next hop type: Router
                                Next hop: 10.4.20.20 via ge-0/0/5.0
                                Session Id: 0x140
                                10.20.20.20/32 Originating RIB: inet.0
                                  Node path count: 1
                                  Forwarding nexthops: 1
                                        Next hop type: Router
```

```
                                          Next hop: 10.4.20.20 via ge-0/0/5.0
                                          Session Id: 0x140
```

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 172.16.5.254 via fxp0.0
10.0.0.1/32         *[Direct/0] 00:51:58
                    > via at-5/3/0.0
10.0.0.2/32         *[Local/0] 00:51:58
                     Local
10.12.12.21/32     *[Local/0] 00:51:57
                     Reject
10.13.13.13/32     *[Direct/0] 00:51:58
                    > via t3-5/2/1.0
10.13.13.14/32     *[Local/0] 00:51:58
                     Local
10.13.13.21/32     *[Local/0] 00:51:58
                     Local
10.13.13.22/32     *[Direct/0] 00:33:59
                    > via t3-5/2/0.0
127.0.0.1/32        [Direct/0] 00:51:58
                    > via lo0.0
10.222.5.0/24     *[Direct/0] 00:51:58
                    > via fxp0.0
10.222.5.81/32    *[Local/0] 00:51:58
                     Local
```

show route table inet.3

```
user@host> show route table inet.3
inet.3: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32        *[LDP/9] 00:25:43, metric 10, tag 200
```

```
                      to 10.2.94.2 via lt-1/2/0.49
                    > to 10.2.3.2 via lt-1/2/0.23
```

show route table inet.3 protocol ospf

```
user@host> show route table inet.3 protocol ospf
inet.3: 9 destinations, 18 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1.1.1.20/32         [L-OSPF/10] 1d 00:00:56, metric 2
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800020
                      to 10.0.6.60 via lt-1/2/0.12, Push 800020, Push 800030(top)
1.1.1.30/32         [L-OSPF/10] 1d 00:01:01, metric 3
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800030
                      to 10.0.6.60 via lt-1/2/0.12, Push 800030
1.1.1.40/32         [L-OSPF/10] 1d 00:01:01, metric 4
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800040
                      to 10.0.6.60 via lt-1/2/0.12, Push 800040
1.1.1.50/32         [L-OSPF/10] 1d 00:01:01, metric 5
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800050
                      to 10.0.6.60 via lt-1/2/0.12, Push 800050
1.1.1.60/32         [L-OSPF/10] 1d 00:01:01, metric 6
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800060
                      to 10.0.6.60 via lt-1/2/0.12, Pop
```

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 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 lsdist.0

```
user@host> show route table lsdist.0

lsdist.0: 21 destinations, 21 routes (21 active, 0 holddown, 0 hidden)
```

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

NODE { AS:100 ISO:0003.0003.0303.00 ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
NODE { AS:100 ISO:1282.0404.9202.00 ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
NODE { AS:100 ISO:1282.0404.9202.02 ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
LINK { Local { AS:100 ISO:0003.0003.0303.00 }.{ IPv4:23.0.0.2 IPv6:23::2 } Remote { AS:100
ISO:1282.0404.9202.02 }.{ } ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
LINK { Local { AS:100 ISO:1282.0404.9202.00 }.{ IPv4:23.0.0.1 IPv6:23::1 } Remote { AS:100
ISO:1282.0404.9202.02 }.{ } ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
LINK { Local { AS:100 ISO:1282.0404.9202.02 }.{ } Remote { AS:100 ISO:0003.0003.0303.00 }.{ }
ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
LINK { Local { AS:100 ISO:1282.0404.9202.02 }.{ } Remote { AS:100 ISO:1282.0404.9202.00 }.{ }
ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:3.3.3.3/32 } ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:32.32.32.32/32 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:33.33.33.33/32 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:100.100.100.0/32 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:128.204.49.193/32 } ISIS-L2:0 }/
```

```
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv4:2.2.2.2/32 } ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv4:128.204.49.202/32 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv6:3::3/128 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv6:23::/64 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv6:100:100:100::100/128 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv6:abcd::128:204:49:193/128 } ISIS-
L2:0 }/1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv6:2::2/128 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv6:23::/64 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv6:abcd::128:204:49:202/128 } ISIS-
L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
```

show route table lsdist.0 (BGP Confederation Enabled)

```
user@host>show route table lsdist.0 detail
lsdist.0: 72 destinations, 72 routes (72 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

NODE { AS:17 Member-ASN:65002 ISO:0004.0404.0400.00 ISIS-L2:0 }/1216
                   *[BGP/170] 00:00:23, localpref 100
                     AS path: (65002) I, validation-state: unverified
                    >  to 11.14.1.2 via ge-0/0/0.0
NODE { AS:17 Member-ASN:65002 ISO:0005.0505.0500.00 ISIS-L2:0 }/1216
                   *[BGP/170] 00:00:23, localpref 100
                      AS path: (65002) I, validation-state: unverified
                    >  to 11.14.1.2 via ge-0/0/0.0

LINK { Local { AS:17 Member-ASN:65001 ISO:0003.0303.0300.00 }.{ IPv4:13.14.1.1
IPv6:abcd::13:14:1:1 } Remote { AS:17 Member-ASN:65001 ISO:0004.0404.0400.00 }.
{ IPv4:13.14.1.2 } ISIS-L2:0 }/1216
                   *[IS-IS/18] 00:00:48
                       Fictitious
LINK { Local { AS:17 Member-ASN:65001 ISO:0003.0303.0300.00 }.{ IPv4:13.14.2.1
IPv6:abcd::13:14:2:1 } Remote { AS:17 Member-ASN:65001 ISO:0004.0404.0400.00 }.
{ IPv4:13.14.2.2 } ISIS-L2:0 }/1216
                   *[IS-IS/18] 00:00:48
                       Fictitious
```

show route table lsdist.0 detail

```
user@host> show route table lsdist.0 detail
lsdist.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)
NODE { AS:200 ISO:1282.2113.1154.00 ISIS-L1:0 }/1216 (1 entry, 1 announced)
*IS-IS  Preference: 15
Level: 1
Next hop type: Fictitious, Next hop index: 0
Address: 0xc5b3054
Next-hop reference count: 14
…………..
…………..
………….
```

```
Area membership:
47 00 05 80 ff f8 00 00 00 01 08 00 01
SPRING-Capabilities:
- SRGB block [Start: 800000, Range: 4096, Flags: 0xc0]
SPRING-Algorithms:
- Algo: 0
SPRING Flex-Algorithms Definition:
- Flex-Algo: 129
Metric: 0, Calc: 0, priority: 129
- Flags: 0x02, - Inc Any: 0x00040000, - Exclude: 0x00008000, - Inc All: 0x00004000
………….
………….
………….
PREFIX { Node { AS:200 ISO:1282.2113.3158.00 } { IPv4:128.220.13.196/32 } ISIS-L1:0 }/1216 (1
entry, 1 announced)
*IS-IS  Preference: 15
Level: 1
Next hop type: Fictitious, Next hop index: 0
Address: 0xc5b3054
Next-hop reference count: 14
Next hop:
State: <Active NotInstall>
Local AS:   200
Age: 16:16:25
………….
………….
Prefix SID: 10, Flags: 0xe0, Algo: 0
Prefix SID: 780, Flags: 0xe0, Algo: 129
Flex Algo: 129, Flex Algo Metric: 10
………….
………….
IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:4000::/64 } ISIS-L1:0 }/1216 (1
entry, 1 announced)
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 588
                Address: 0x7660f64
                Next-hop reference count: 12, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 21.0.2.1
                Next hop: 21.0.2.1 via ge-0/0/0.0, selected
                Session Id: 320
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 1d 10:20:38
```

```
                Validation State: unverified
                Task: BGP_100.21.0.2.1
                Announcement bits (1): 0-TED Export
                AS path: 100 I
                Accepted
                SRv6 Locator Flags: 0, Algo: 0, Metric: 0
                Localpref: 100
                Router ID: 100.100.100.0
                Thread: junos-main
```

show route table lsdist.0 detail

```
user@host> show route table lsdist.1 detail
SPRING Flex-Algorithms Definition:
                 - Flex-Algo: 128
                   Metric: 2, Calc: 1, Priority: 10
                    - Flags: 0X80000000, - Inc Any: 0x00000002, - Exclude: 0x00000004, - Inc 
All: 0x00000002
```

show route table lsdist.0 extensive

The output of the `show route table lsdist.0 extensive` command to include IPv6 Prefix attributes.

```
user@host> show route table lsdist.0 extensive
lsdist.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
NODE { AS:100 ISO:0100.0100.0100.00 ISIS-L1:0 }/1216 (1 entry, 1 announced)
TSI:
LINK-STATE attribute handle 0x75ace70
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 0
                Address: 0x7661124
                Next-hop reference count: 5, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 21.0.2.1
                Next hop: 21.0.2.1 via ge-0/0/0.0, selected
                Session Id: 0
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 17
                Validation State: unverified
```

```
                Task: BGP_100.21.0.2.1
                Announcement bits (1): 0-TED Export
                AS path: 100 I
                Accepted
                IPv4 Router-ids:
                  100.100.100.0
                Area border router: No
                External router: No
                Attached: No
                Overload: No
                Hostname: R0
                Area membership:
                  49 00 05
                SPRING-Algorithms:
                 - Algo: 0
                 - Algo: 1
                SRV6 Capable: - Flags: 0
                SRV6 Node MSD:
                 - Type: 41, Value: 6
                 - Type: 42, Value: 7
                 - Type: 43, Value: 5
                 - Type: 44, Value: 6
                 - Type: 45, Value: 6
                Localpref: 100
                Router ID: 100.100.100.0
                Thread: junos-main

PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv4:100.100.100.0/32 } ISIS-L1:0 }/1216 (1
entry, 1 announced)
TSI:
LINK-STATE attribute handle 0x0
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 0
                Address: 0x7661124
                Next-hop reference count: 5, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 21.0.2.1
                Next hop: 21.0.2.1 via ge-0/0/0.0, selected
                Session Id: 0
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 28
                Validation State: unverified
                Task: BGP_100.21.0.2.1
```

```
                Announcement bits (1): 0-TED Export
                AS path: 100 I
                Accepted
                Localpref: 100
                Router ID: 100.100.100.0
                Thread: junos-main

IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:10::10/128 } ISIS-L1:0 }/1216 (1
entry, 1 announced)
TSI:
LINK-STATE attribute handle 0x0
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 0
                Address: 0x7661124
                Next-hop reference count: 5, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 21.0.2.1
                Next hop: 21.0.2.1 via ge-0/0/0.0, selected
                Session Id: 0
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 28
                Validation State: unverified
                Task: BGP_100.21.0.2.1
                Announcement bits (1): 0-TED Export
                AS path: 100 I
                Accepted
                Localpref: 100
                Router ID: 100.100.100.0
                Thread: junos-main
```

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.0 protocol ospf

```
user@host> show route table mpls.0 protocol ospf
mpls.0: 29 destinations, 29 routes (29 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

299952               *[L-OSPF/10] 23:59:42, metric 0
                      > to 10.0.10.70 via lt-1/2/0.14, Pop
                        to 10.0.6.60 via lt-1/2/0.12, Swap 800070, Push 800030(top)
299952(S=0)          *[L-OSPF/10] 23:59:42, metric 0
                      > to 10.0.10.70 via lt-1/2/0.14, Pop
                        to 10.0.6.60 via lt-1/2/0.12, Swap 800070, Push 800030(top)
299968               *[L-OSPF/10] 23:59:48, metric 0
                      > to 10.0.6.60 via lt-1/2/0.12, Pop
```

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
```

Release Information

Command introduced before Junos OS Release 7.4.

Show route table evpn statement introduced in Junos OS Release 15.1X53-D30 for QFX Series switches.

RELATED DOCUMENTATION

show route summary

show route terse

IN THIS SECTION

Syntax

```
show route terse
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route terse
```

Description

Display a high-level summary of the routes in the routing table.

NOTE: For BGP routes, the `show route terse` command displays the local preference attribute and MED instead of the metric1 and metric2 values. This is mostly due to historical reasons.

To display the metric1 and metric2 value of a BGP route, use the `show route extensive` command.

Options

none	Display a high-level summary of the routes in the routing table.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 28 on page 556 describes the output fields for the `show route terse` command. Output fields are listed in the approximate order in which they appear.

Table 28: show route terse Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).
number destinations	Number of destinations for which there are routes in the routing table.
number routes	Number of routes in the routing table and total number of routes in the following states: • `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)
route key	Key for the state of the route: • `+`—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.
`A`	Active route. An asterisk (*) indicates this is the active route.

Table 28: show route terse Output Fields *(Continued)*

Field Name	Field Description
`V`	Validation status of the route: • `?`—Not evaluated. Indicates that the route was not learned through BGP. • `I`—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. • `N`—Unknown. Indicates that the prefix is not among the prefixes or prefix ranges in the database. • `V`—Valid. Indicates that the prefix and autonomous system pair are found in the database.
Destination	Destination of the route.

Table 28: show route terse Output Fields *(Continued)*

Field Name	Field Description
P	Protocol through which the route was learned: • `A`—Aggregate • `B`—BGP • `C`—CCC • `D`—Direct • `G`—GMPLS • `I`—IS-IS • `L`—L2CKT, L2VPN, LDP, Local • `K`—Kernel • `M`—MPLS, MSDP • `O`—OSPF • `P`—PIM • `R`—RIP, RIPng • `S`—Static • `T`—Tunnel
Prf	Preference value of the route. 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.
Metric 1	First metric value in the route. For routes learned from BGP, this is the MED metric.
Metric 2	Second metric value in the route. For routes learned from BGP, this is the IGP metric.

Table 28: show route terse Output Fields *(Continued)*

Field Name	Field Description
Next hop	Next hop to the destination. An angle bracket (>) indicates that the route is the selected route.
AS path	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: • `I`—IGP. • `E`—EGP. • `?`—Incomplete; typically, the AS path was aggregated.

Sample Output

show route terse

```
user@host> show route terse
inet.0: 10 destinations, 12 routes (10 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A V Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* ? 172.16.1.1/32        O  10         1             >10.0.0.2
  ?                    B 170        100                             I
  unverified                                      >10.0.0.2
* ? 172.16.1.1/32        D   0                       >lo0.2
* V 2.2.0.2/32         B 170        110                             200 I
  valid                                           >10.0.0.2
* ? 10.0.0.0/30        D   0                      >lt-1/2/0.1
  ?                    B 170        100                             I
  unverified                                      >10.0.0.2
* ? 10.0.0.1/32        L   0                        Local
* ? 10.0.0.4/30        B 170        100                             I
  unverified                                      >10.0.0.2
* ? 10.0.0.8/30        B 170        100                             I
```

```
  unverified                                    >10.0.0.2
* I 172.16.1.1/32      B 170         90                         200 I
  invalid                                       >10.0.0.2
* N 192.168.2.3/32     B 170        100                         200 I
  unknown                                       >10.0.0.2
* ? 172.16.233.5/32       O  10          1          MultiRecv
```

Release Information

Command introduced before Junos OS Release 7.4.

test policy

IN THIS SECTION

Syntax

```
test policy policy-name prefix
```

Description

Test a policy configuration to determine which prefixes match routes in the routing table.

> **NOTE**: If you are using the `test policy` command on a logical system, you must first set the CLI to the logical system context. For example, if you want to test a routing policy that is configured on logical system R2, first run the `set cli logical-system R2` command.

Options

policy-name	Name of a policy.
prefix	Destination prefix to match.

Additional Information

All prefixes in the default unicast routing table (inet.0) that match prefixes that are the same as or longer than the specific prefix are processed by the `from` clause in the specified policy. All prefixes accepted by the policy are displayed. The `test policy` command evaluates a policy differently from the BGP import process. When testing a policy that contains an `interface` match condition in the `from` clause, the `test policy` command uses the match condition. In contrast, BGP does not use the `interface` match condition when evaluating the policy against routes learned from internal BGP (IBGP) or external BGP (EGBP) multihop peers.

When testing a policy, you can see the length of time (in microseconds) required to evaluate the policy and the number of times it has been executed by running the `show policy` *`policy-name`* `statistics` command.

Required Privilege Level

view

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

test policy

```
user@host> test policy test-statics 172.16.0.1/8
inet.0: 44 destinations, 44 routes (44 active, 0 holddown, 0 hidden)
Prefixes passing policy:

172.16.3.0/8          *[BGP/170] 16:22:46, localpref 100, from 10.255.255.41
                   AS Path: 50888 I
                    > to 10.11.4.32 via en0.2, label-switched-path l2
172.16.3.1/32         *[IS-IS/18] 2d 00:21:46, metric 0, tag 2
                    > to 10.0.4.7 via fxp0.0
172.16.3.2/32         *[IS-IS/18] 2d 00:21:46, metric 0, tag 2
                    > to 10.0.4.7 via fxp0.0
172.16.3.3/32         *[IS-IS/18] 2d 00:21:46, metric 0, tag 2
                    > to 10.0.4.7 via fxp0.0
172.16.3.4/32         *[IS-IS/18] 2d 00:21:46, metric 0, tag 2
                    > to 10.0.4.7 via fxp0.0
Policy test-statics: 5 prefixes accepted, 0 prefixes rejected
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Understanding Routing Policy Tests

show policy

show route

show route detail

show route extensive

show route terse

8

CHAPTER

RIPng Operational Commands

clear ripng general-statistics

IN THIS SECTION

Syntax

```
clear ripng general-statistics
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
clear ripng general-statistics
```

Description

Clear RIP next generation (RIPng) general statistics.

Options

none	Clear RIPng general statistics.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

clear

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear ripng general-statistics

```
user@host> clear ripng general-statistics
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show ripng general-statistics | 593

clear ripng statistics

IN THIS SECTION

Syntax

```
clear ripng statistics
<instance    | name>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switch)

```
clear ripng statistics
<instance    | name>
```

Description

Clear RIP next-generation (RIPng) statistics.

Options

none	Reset RIPng counters for all neighbors for all routing instances.
instance	(Optional) Reset RIPng counters for the specified instance.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
name	(Optional) Reset RIPng counters for the specified neighbor.

Required Privilege Level

clear

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear ripng statistics

```
user@host> clear ripng statistics
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show ripng statistics | 598

restart

IN THIS SECTION

Syntax

```
restart
<adaptive-services |ancpd-service | application-identification |audit-process | auto-
configuration |captive-portal-content-delivery |ce-l2tp-service | chassis-control  | class-of-
service |clksyncd-service |database-replication|datapath-trace-service |dhcp-service | diameter-
```

```
service | disk-monitoring | dynamic-flow-capture | ecc-error-logging | ethernet-connectivity-
fault-management |ethernet-link-fault-management |event-processing | firewall | general-
authentication-service | gracefully | iccp-service |idp-policy | immediately |interface-control
| ipsec-key-management | kernel-health-monitoring | kernel-replication | l2-learning | l2cpd-
service | l2tp-service | l2tp-universal-edge | lacp | license-service |link-management |local-
policy-decision-function |mac-validation |mib-process | mountd-service |mpls-traceroute |mspd |
multicast-snooping |named-service | nfsd-service | packet-triggered-subscribers |peer-selection-
service |pgm | pic-services-logging | pki-service |ppp | ppp-service | pppoe | protected-system-
domain-service | redundancy-interface-process | remote-operations | root-system-domain-service |
routing <logical-system logical-system-name>  | sampling | sbc-configuration-process | sdk-
service |service-deployment | services | snmp |soft |static-subscribers |statistics-service|
subscriber-management | subscriber-management-helper | tunnel-oamd |usb-control| vrrp |web-
management>
<gracefully | immediately | soft>
```

Syntax (ACX Series Routers)

```
restart
<adaptive-services |audit-process | auto-configuration | autoinstallation |chassis-control |
class-of-service |clksyncd-service |database-replication| dhcp-service | diameter-service | disk-
monitoring | dynamic-flow-capture | ethernet-connectivity-fault-management | ethernet-link-fault-
management |event-processing | firewall | general-authentication-service | gracefully |
immediately |interface-control | ipsec-key-management | l2-learning | lacp |link-management |mib-
process | mountd-service |mpls-traceroute |mspd | named-service | nfsd-service | pgm | pki-
service | ppp | pppoe | redundancy-interface-process | remote-operations | routing | sampling |
sdk-service |secure-neighbor-discovery | service-deployment | services | snmp |soft | statistics-
service| subscriber-management | subscriber-management-helper | tunnel-oamd | vrrp>
```

Syntax (EX Series Switches)

```
restart
<autoinstallation | chassis-control | class-of-service | database-replication | dhcp | dhcp-
service | diameter-service | dot1x-protocol | ethernet-link-fault-management | ethernet-
switching | event-processing | firewall | general-authentication-service | interface-control |
kernel-health-monitoring | kernel-replication | l2-learning | lacp | license-service | link-
management | lldpd-service | mib-process | mountd-service | multicast-snooping | pgm |
```

```
redundancy-interface-process | remote-operations | routing | secure-neighbor-discovery | service-
deployment | sflow-service | snmp | vrrp | web-management>
```

Syntax (MX Series Routers)

```
restart
<adaptive-services | ancpd-service | application-identification | audit-process | auto-
configuration | bbe-stats-service | captive-portal-content-delivery | ce-l2tp-service | chassis-
control | class-of-service | clksyncd-service | database-replication | datapath-trace-service |
dhcp-service | diameter-service | disk-monitoring | dynamic-flow-capture | ecc-error-logging |
ethernet-connectivity-fault-management | ethernet-link-fault-management | event-processing |
firewall | general-authentication-service | gracefully | iccp-service | idp-policy | immediately
|interface-control | ipsec-key-management |kernel-health-monitoring |  kernel-replication | l2-
learning | l2cpd-service | l2tp-service | l2tp-universal-edge | lacp | license-service | link-
management | local-policy-decision-function | mac-validation | mib-process | mountd-service |
mpls-traceroute | mspd | multicast-snooping |named-service | nfsd-service | packet-triggered-
subscribers |peer-selection-service | pgm | pic-services-logging | pki-service | ppp | ppp-
service | pppoe | protected-system-domain-service | redundancy-interface-process | remote-
operations | root-system-domain-service | routing | routing <logical-system logical-system-
name>  | sampling | sbc-configuration-process | sdk-service | service-deployment | services |
snmp |soft |static-subscribers |statistics-service| subscriber-management | subscriber-
management-helper | tunnel-oamd | usb-control | vrrp | web-management>
<all-members>
<gracefully | immediately | soft>
<local>
<member member-id>
```

Syntax (QFX Series)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dialer-services |
diameter-service | dlsw | ethernet-connectivity | event-processing | fibre-channel | firewall |
general-authentication-service | igmp-host-services | interface-control | ipsec-key-management |
isdn-signaling | l2ald | l2-learning | l2tp-service | mib-process | named-service | network-
access-service | nstrace-process | pgm | ppp | pppoe | redundancy-interface-process | remote-
operations |logical-system-name>  | routing | sampling |secure-neighbor-discovery | service-
```

```
deployment | snmp | usb-control | web-management>
<gracefully | immediately | soft>
```

Syntax (Routing Matrix)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | disk-monitoring |
dynamic-flow-capture | ecc-error-logging | event-processing | firewall | interface-control |
ipsec-key-management | kernel-replication | l2-learning | l2tp-service | lacp | link-management
| mib-process | pgm | pic-services-logging | ppp | pppoe | redundancy-interface-process | remote-
operations | routing <logical-system logical-system-name>  | sampling | service-deployment |
snmp>
<all | all-lcc | lcc number>
<gracefully | immediately | soft>
```

Syntax (SRX Series)

```
restart
<application-identification |application-security |audit-process |commitd-service |chassis-
control | class-of-service |database-replication |datapath-trace-service |ddns |dhcp |dhcp-
service |dynamic-flow-capture |disk-monitoring |event-processing | ethernet-connectivity-fault-
management |ethernet-link-fault-management |extensible-subscriber-services |fipsd |firewall |
firewall-authentication-service |general-authentication-service |gracefully |gprs-process |idp-
policy |immediately |interface-control | ipmi |ipsec-key-management |jflow-service |jnu-
management |jnx-wmicd-service |jsrp-service |kernel-replication |l2-learning |l2cpd-service |
lacp |license-service |logical-system-service |mib-process |mountd-service |named-service |
network-security |network-security-trace |nfsd-service |ntpd-service |pgm |pic-services-logging |
profilerd |pki-service |remote-operations |rest-api |routing |sampling |sampling-route-record |
scc-chassisd |secure-neighbor-discovery |security-intelligence |security-log |services |service-
deployment |simple-mail-client-service |soft |snmp |static-routed |statistics-service |
subscriber-management |subscriber-management-helper |system-log-vital |tunnel-oamd |uac-service |
user-ad-authentication |vrrp |web-management >
```

Syntax (TX Matrix Routers)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dhcp-service | diameter-
service | disk-monitoring | dynamic-flow-capture | ecc-error-logging | event-processing |
firewall | interface-control | ipsec-key-management | kernel-replication | l2-learning | l2tp-
service | lacp | link-management | mib-process |pgm | pic-services-logging | ppp | pppoe |
redundancy-interface-process | remote-operations | routing <logical-system logical-system-name>
| sampling | service-deployment | snmp| statistics-service>
<all-chassis | all-lcc | lcc number | scc>
<gracefully | immediately | soft>
```

Syntax (TX Matrix Plus Routers)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dhcp-service | diameter-
service | disk-monitoring | dynamic-flow-capture | ecc-error-logging | event-processing |
firewall | interface-control | ipsec-key-management | kernel-replication | l2-learning | l2tp-
service | lacp | link-management | mib-process | pgm | pic-services-logging | ppp | pppoe |
redundancy-interface-process | remote-operations | routing <logical-system logical-system-name>
| sampling | service-deployment | snmp| statistics-service>
<all-chassis | all-lcc | all-sfc | lcc number | sfc number>
<gracefully | immediately | soft>
```

Syntax (QFX Series)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dialer-services |
diameter-service | dlsw | ethernet-connectivity | event-processing | fibre-channel | firewall |
general-authentication-service | igmp-host-services | interface-control | ipsec-key-management |
isdn-signaling | l2ald | l2-learning | l2tp-service | mib-process | named-service | network-
access-service | nstrace-process | pgm | ppp | pppoe | redundancy-interface-process | remote-
operations |logical-system-name>  | routing | sampling |secure-neighbor-discovery | service-
```

```
deployment | snmp | usb-control | web-management>
<gracefully | immediately | soft>
```

Description

Restart a Junos OS process.

> **CAUTION**: Never restart a software process unless instructed to do so by a customer support engineer. A restart might cause the router or switch to drop calls and interrupt transmission, resulting in possible loss of data.

The `restart` command expands all applications names including applications that are not required for the current platform. Therefore, a user could try to do a restart for an application that is not running for the current platform. This error message communicates that the restart failed because the application was not running on the system.

Options

none	Same as `gracefully`.
adaptive-services	(Optional) Restart the configuration management process that manages the configuration for stateful firewall, Network Address Translation (NAT), intrusion detection services (IDS), and IP Security (IPsec) services on the Adaptive Services PIC.
all-chassis	(TX Matrix and TX Matrix Plus routers only) (Optional) Restart the software process on all chassis.
all-lcc	(TX Matrix and TX Matrix Plus routers only) (Optional) For a TX Matrix router, restart the software process on all T640 routers connected to the TX Matrix router. For a TX Matrix Plus router, restart the software process on all T1600 routers connected to the TX Matrix Plus router.
all-members	(MX Series routers only) (Optional) Restart the software process for all members of the Virtual Chassis configuration.

all-sfc	(TX Matrix Plus routers only) (Optional) For a TX Matrix Plus router, restart the software processes for the TX Matrix Plus router (or switch-fabric chassis).
ancpd-service	(Optional) Restart the Access Node Control Protocol (ANCP) process, which works with a special Internet Group Management Protocol (IGMP) session to collect outgoing interface mapping events in a scalable manner.
application-identification	(Optional) Restart the process that identifies an application using intrusion detection and prevention (IDP) to allow or deny traffic based on applications running on standard or nonstandard ports.
application-security	(Optional) Restart the application security process.
audit-process	(Optional) Restart the RADIUS accounting process that gathers statistical data that can be used for general network monitoring, analyzing, and tracking usage patterns, for billing a user based on the amount of time or type of services accessed.
auto-configuration	(Optional) Restart the Interface Auto-Configuration process.
autoinstallation	(EX Series switches only) (Optional) Restart the autoinstallation process.
bbe-stats-service	(MX Series routers only) (Optional) Restart bbe-statsd, the BBE statistics collection and management process.
captive-portal-content-delivery	(Optional) Restart the HTTP redirect service by specifying the location to which a subscriber's initial Web browser session is redirected, enabling initial provisioning and service selection for the subscriber.
ce-l2tp-service	(M10, M10i, M7i, and MX Series routers only) (Optional) Restart the Universal Edge Layer 2 Tunneling Protocol (L2TP) process, which establishes L2TP tunnels and Point-to-Point Protocol (PPP) sessions through L2TP tunnels.
chassis-control	(Optional) Restart the chassis management process.
class-of-service	(Optional) Restart the class-of-service (CoS) process, which controls the router's or switch's CoS configuration.
clksyncd-service	(Optional) Restart the external clock synchronization process, which uses synchronous Ethernet (SyncE).
commitd-service	(Optional) Restart the committed services.
database-replication	(EX Series switches and MX Series routers only) (Optional) Restart the database replication process.

datapath-trace-service	(Optional) Restart the packet path tracing process.
dhcp	(EX Series switches only) (Optional) Restart the software process for a Dynamic Host Configuration Protocol (DHCP) server. A DHCP server allocates network IP addresses and delivers configuration settings to client hosts without user intervention.
dhcp-service	(Optional) Restart the Dynamic Host Configuration Protocol process.
dialer-services	(EX Series switches only) (Optional) Restart the ISDN dial-out process.
diameter-service	(Optional) Restart the diameter process.
disk-monitoring	(Optional) Restart disk monitoring, which checks the health of the hard disk drive on the Routing Engine.
dlsw	(QFX Series only) (Optional) Restart the data link switching (DLSw) service.
dot1x-protocol	(EX Series switches only) (Optional) Restart the port-based network access control process.
dynamic-flow-capture	(Optional) Restart the dynamic flow capture (DFC) process, which controls DFC configurations on Monitoring Services III PICs.
ecc-error-logging	(Optional) Restart the error checking and correction (ECC) process, which logs ECC parity errors in memory on the Routing Engine.
ethernet-connectivity-fault-management	(Optional) Restart the process that provides IEEE 802.1ag Operation, Administration, and Management (OAM) connectivity fault management (CFM) database information for CFM maintenance association end points (MEPs) in a CFM session.
ethernet-link-fault-management	(EX Series switches and MX Series routers only) (Optional) Restart the process that provides the OAM link fault management (LFM) information for Ethernet interfaces.
ethernet-switching	(EX Series switches only) (Optional) Restart the Ethernet switching process.
event-processing	(Optional) Restart the event process (eventd).
extensible-subscriber-services	(Optional) Restart the extensible subscriber services process.
fibre-channel	(QFX Series only) (Optional) Restart the Fibre Channel process.
fipsd	(Optional) Restart the fipsd services.

firewall	(Optional) Restart the firewall management process, which manages the firewall configuration and enables accepting or rejecting packets that are transiting an interface on a router or switch.
general-authentication-service	(EX Series switches and MX Series routers only) (Optional) Restart the general authentication process.
gprs-process	(Optional) Restart the General Packet Radio Service (GPRS) process.
gracefully	(Optional) Restart the software process.
iccp-service	(Optional) Restart the Inter-Chassis Communication Protocol (ICCP) process.
idp-policy	(Optional) Restart the intrusion detection and prevention (IDP) protocol process.
immediately	(Optional) Immediately restart the software process.
interface-control	(Optional) Restart the interface process, which controls the router's or switch's physical interface devices and logical interfaces.
ipmi	(Optional) Restart the intelligent platform management interface process.
ipsec-key-management	(Optional) Restart the IPsec key management process.
isdn-signaling	(QFX Series only) (Optional) Restart the ISDN signaling process, which initiates ISDN connections.
jflow-service	(Optional) Restart jflow service process.
jnu-management	(Optional) Restart jnu management process.
jnx-wmicd-service	(Optional) Restart jnx wmicd service process.
jsrp-service	(Optional) Restart the Juniper Services Redundancy Protocol (jsrdp) process, which controls chassis clustering.
kernel-health-monitoring	(Optional) Restart the Routing Engine kernel health monitoring process, which enables health parameter data to be sent from kernel components to data collection applications. When you change the polling interval through `sysctl kern.jkhmd_polling_time_secs`, you must restart the kernel health monitoring process for the new polling interval to take effect.
kernel-replication	(Optional) Restart the kernel replication process, which replicates the state of the backup Routing Engine when graceful Routing Engine switchover (GRES) is configured.

l2-learning	(Optional) Restart the Layer 2 address flooding and learning process.
l2cpd-service	(Optional) Restart the Layer 2 Control Protocol process, which enables features such as Layer 2 protocol tunneling and nonstop bridging.
l2tp-service	(M10, M10i, M7i, and MX Series routers only) (Optional) Restart the Layer 2 Tunneling Protocol (L2TP) process, which sets up client services for establishing Point-to-Point Protocol (PPP) tunnels across a network and negotiating Multilink PPP if it is implemented.
l2tp-universal-edge	(MX Series routers only) (Optional) Restart the L2TP process, which establishes L2TP tunnels and PPP sessions through L2TP tunnels.
lacp	(Optional) Restart the Link Aggregation Control Protocol (LACP) process. LACP provides a standardized means for exchanging information between partner systems on a link to allow their link aggregation control instances to reach agreement on the identity of the LAG to which the link belongs, and then to move the link to that LAG, and to enable the transmission and reception processes for the link to function in an orderly manner.
lcc *number*	(TX Matrix and TX Matrix Plus routers only) (Optional) For a TX Matrix router, restart the software process for a specific T640 router that is connected to the TX Matrix router. For a TX Matrix Plus router, restart the software process for a specific router that is connected to the TX Matrix Plus router. Replace *number* with the following values depending on the LCC configuration: • 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix. • 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix. • 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix. • 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
license-service	(EX Series switches only) (Optional) Restart the feature license management process.
link-management	(TX Matrix and TX Matrix Plus routers and EX Series switches only) (Optional) Restart the Link Management Protocol (LMP) process, which establishes and maintains LMP control channels.

lldpd-service	(EX Series switches only) (Optional) Restart the Link Layer Discovery Protocol (LLDP) process.
local	(MX Series routers only) (Optional) Restart the software process for the local Virtual Chassis member.
local-policy-decision-function	(Optional) Restart the process for the Local Policy Decision Function, which regulates collection of statistics related to applications and application groups and tracking of information about dynamic subscribers and static interfaces.
logical-system-service	(Optional) Restart the logical system service process.
mac-validation	(Optional) Restart the Media Access Control (MAC) validation process, which configures MAC address validation for subscriber interfaces created on demux interfaces in dynamic profiles on MX Series routers.
member *member-id*	(MX Series routers only) (Optional) Restart the software process for a specific member of the Virtual Chassis configuration. Replace *member-id* with a value of `0` or `1`.
mib-process	(Optional) Restart the Management Information Base (MIB) version II process, which provides the router's MIB II agent.
mobile-ip	(Optional) Restart the Mobile IP process, which configures Junos OS Mobile IP features.
mountd-service	(EX Series switches and MX Series routers only) (Optional) Restart the service for NFS mount requests.
mpls-traceroute	(Optional) Restart the MPLS Periodic Traceroute process.
mspd	(Optional) Restart the Multiservice process.
multicast-snooping	(EX Series switches and MX Series routers only) (Optional) Restart the multicast snooping process, which makes Layer 2 devices, such as VLAN switches, aware of Layer 3 information, such as the media access control (MAC) addresses of members of a multicast group.
named-service	(Optional) Restart the DNS Server process, which is used by a router or a switch to resolve hostnames into addresses.
network-access-service	(QFX Series only) (Optional) Restart the network access process, which provides the router's Challenge Handshake Authentication Protocol (CHAP) authentication service.

network-security	(Optional) Restart the network security process.
network-security-trace	(Optional) Restart the network security trace process.
nfsd-service	(Optional) Restart the Remote NFS Server process, which provides remote file access for applications that need NFS-based transport.
ntpd-service	(Optional) Restart the Network Time Protocol (NTP) process.
packet-triggered-subscribers	(Optional) Restart the packet-triggered subscribers and policy control (PTSP) process, which allows the application of policies to dynamic subscribers that are controlled by a subscriber termination device.
peer-selection-service	(Optional) Restart the Peer Selection Service process.
pgcp-service	(Optional) Restart the pgcpd service process running on the Routing Engine. This option does not restart pgcpd processes running on mobile station PICs. To restart pgcpd processes running on mobile station PICs, use the `services pgcp gateway` option.
pgm	(Optional) Restart the process that implements the Pragmatic General Multicast (PGM) protocol for assisting in the reliable delivery of multicast packets.
pic-services-logging	(Optional) Restart the logging process for some PICs. With this process, also known as fsad (the file system access daemon), PICs send special logging information to the Routing Engine for archiving on the hard disk.
pki-service	(Optional) Restart the PKI Service process.
ppp	(Optional) Restart the Point-to-Point Protocol (PPP) process, which is the encapsulation protocol process for transporting IP traffic across point-to-point links.
ppp-service	(Optional) Restart the Universal edge PPP process, which is the encapsulation protocol process for transporting IP traffic across universal edge routers.
pppoe	(Optional) Restart the Point-to-Point Protocol over Ethernet (PPPoE) process, which combines PPP that typically runs over broadband connections with the Ethernet link-layer protocol that allows users to connect to a network of hosts over a bridge or access concentrator.
profilerd	(Optional) Restart the profiler process.
protected-system-domain-service	(Optional) Restart the Protected System Domain (PSD) process.

redundancy-interface-process	(Optional) Restart the ASP redundancy process.
remote-operations	(Optional) Restart the remote operations process, which provides the ping and traceroute MIBs.
rest-api	(Optional) Restart the rest api process.
root-system-domain-service	(Optional) Restart the Root System Domain (RSD) service.
routing	(ACX Series routers, QFX Series, EX Series switches, and MX Series routers only) (Optional) Restart the routing protocol process.
routing <logical-system *logical-system-name*>	(Optional) Restart the routing protocol process, which controls the routing protocols that run on the router or switch and maintains the routing tables. Optionally, restart the routing protocol process for the specified logical system only.
sampling	(Optional) Restart the sampling process, which performs packet sampling based on particular input interfaces and various fields in the packet header.
sampling-route-record	(Optional) Restart the sampling route record process.
sbc-configuration-process	(Optional) Restart the session border controller (SBC) process of the border signaling gateway (BSG).
scc	(TX Matrix routers only) (Optional) Restart the software process on the TX Matrix router (or switch-card chassis).
scc-chassisd	(Optional) Restart the scc chassisd process.
sdk-service	(Optional) Restart the SDK Service process, which runs on the Routing Engine and is responsible for communications between the SDK application and Junos OS. Although the SDK Service process is present on the router, it is turned off by default.
secure-neighbor-discovery	(QFX Series, EX Series switches, and MX Series routers only) (Optional) Restart the secure Neighbor Discovery Protocol (NDP) process, which provides support for protecting NDP messages.
security-intelligence	(Optional) Restart security intelligence process.
security-log	(Optional) Restart the security log process.

sfc *number*	(TX Matrix Plus routers only) (Optional) Restart the software process on the TX Matrix Plus router (or switch-fabric chassis). Replace `number` with `0`.
service-deployment	(Optional) Restart the service deployment process, which enables Junos OS to work with the Session and Resource Control (SRC) software.
services	(Optional) Restart a service.
services pgcp gateway *gateway-name*	(Optional) Restart the pgcpd process for a specific border gateway function (BGF) running on an MS-PIC. This option does not restart the pgcpd process running on the Routing Engine. To restart the pgcpd process on the Routing Engine, use the `pgcp-service` option.
sflow-service	(EX Series switches only) (Optional) Restart the flow sampling (sFlow technology) process.
simple-mail-client-service	(Optional) Restart the simple mail client service process.
snmp	(Optional) Restart the SNMP process, which enables the monitoring of network devices from a central location and provides the router's or switch's SNMP master agent.
soft	(Optional) Reread and reactivate the configuration without completely restarting the software processes. For example, BGP peers stay up and the routing table stays constant. Omitting this option results in a graceful restart of the software process.
static-routed	(Optional) Restart the static routed process.
static-subscribers	(Optional) Restart the static subscribers process, which associates subscribers with statically configured interfaces and provides dynamic service activation and activation for these subscribers.
statistics-service	(Optional) Restart the process that manages the Packet Forwarding Engine statistics.
subscriber-management	(Optional) Restart the Subscriber Management process.
subscriber-management-helper	(Optional) Restart the Subscriber Management Helper process.
system-log-vital	(Optional) Restart system log vital process.
tunnel-oamd	(Optional) Restart the Tunnel OAM process, which enables the Operations, Administration, and Maintenance of Layer 2 tunneled networks. Layer 2

protocol tunneling (L2PT) allows service providers to send Layer 2 protocol data units (PDUs) across the provider's cloud and deliver them to Juniper Networks EX Series Ethernet Switches that are not part of the local broadcast domain.

uac-service (Optional) Restart the Unified Access Control (UAC) process.

usb-control (MX Series routers) (Optional) Restart the USB control process.

user-ad-authentication (Optional) Restart User ad Authentication process

vrrp (ACX Series routers, EX Series switches, and MX Series routers only) (Optional) Restart the Virtual Router Redundancy Protocol (VRRP) process, which enables hosts on a LAN to make use of redundant routing platforms on that LAN without requiring more than the static configuration of a single default route on the hosts.

web-management (QFX Series, EX Series switches, and MX Series routers only) (Optional) Restart the Web management process.

Required Privilege Level

reset

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

restart interfaces

```
user@host> restart interfaces
interfaces process terminated
interfaces process restarted
```

restart interface-control gracefully

```
user@host> restart interface-control gracefully
Interface control process started, pid 41129
```

restart interface-control (Junos OS Evolved)

```
user@host> restart interface-control
interface-control restart requested
Restarted aggd on re0
Restarted ifmand on re0
```

Release Information

Command introduced before Junos OS Release 7.4.

Options added:

- `dynamic-flow-capture` in Junos OS Release 7.4.
- `dlsw` in Junos OS Release 7.5.
- `event-processing` in Junos OS Release 7.5.
- `ppp` in Junos OS Release 7.5.
- `l2ald` in Junos OS Release 8.0.
- `link-management` in Junos Release 8.0.
- `pgcp-service` in Junos OS Release 8.4.
- `sbc-configuration-process` in Junos OS Release 9.5.
- `services pgcp gateway` in Junos OS Release 9.6.
- `sfc` and `all-sfc` for the TX Matrix Router in Junos OS Release 9.6.
- Command introduced before Junos OS Release 9.2 on SRX Series Firewalls.
- `bbe-stats-service` in Junos OS Release 18.4R1 on MX Series routers.

- `kernel-health-monitoring` in Junos OS Release 19.1R1.
- Introduced in Junos OS Evolved Release 19.1R1.

RELATED DOCUMENTATION

Overview of Operational Mode Commands

show policy

IN THIS SECTION

Syntax

```
show policy
<logical-system (all | logical-system-name)>
<policy-name>
<statistics >
```

Syntax (EX Series Switches)

```
show policy
<policy-name>
```

Description

Display information about configured routing policies.

Options

none	List the names of all configured routing policies.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
policy-name	(Optional) Show the contents of the specified policy.
statistics	(Optional) Use in conjunction with the `test policy` command to show the length of time (in microseconds) required to evaluate a given policy and the number of times it has been executed. This information can be used, for example, to help structure a policy so it is evaluated efficiently. Timers shown are per route; times are not cumulative. Statistics are incremented even when the router is learning (and thus evaluating) routes from peering routers.

Required Privilege Level

view

Output Fields

Table 29 on page 588 lists the output fields for the `show policy` command. Output fields are listed in the approximate order in which they appear.

Table 29: show policy Output Fields

Field Name	Field Description
policy-name	Name of the policy listed.
term	Name of the user-defined policy term. The term name `unnamed` is used for policy elements that occur outside of user defined terms
from	Match condition for the policy.
then	Action for the policy.

Sample Output

show policy

```
user@host> show policy
Configured policies:
__vrf-export-red-internal__
__vrf-import-red-internal__
red-export
rf-test-policy
multicast-scoping
```

show policy policy-name

```
user@host> show policy vrf-import-red-internal
Policy vrf-import-red-internal:
    from
```

```
        203.0.113.0/28  accept
        203.0.113.32/28  accept
    then reject
```

show policy statistics policy-name

```
user@host> show policy statistics iBGP-v4-RR-Import
Policy iBGP-v4-RR-Import:
    [1243328] Term Lab-Infra:
        from [1243328 0]  proto BGP
             [28 0] route filter:
                 10.11.0.0/8 orlonger
                 10.13.0.0/8 orlonger
        then [28 0] accept
    [1243300] Term External:
        from [1243300 1]  proto BGP
         [1243296 0]  community Ext-Com1 [64496:1515 ]
         [1243296 0]  prefix-list-filter Customer-Routes
         [1243296 0]  aspath AS6221
             [1243296 1] route filter:
                 172.16.49.0/12 orlonger
                 172.16.50.0/12 orlonger
                 172.16.51.0/12 orlonger
                 172.16.52.0/12 orlonger
                 172.16.56.0/12 orlonger
                 172.16.60.0/12 orlonger
        then [1243296 2] community + Ext-Com2 [64496:2000 ] [1243296 0] accept
    [4] Term Final:
        then [4 0] reject
```

Release Information

Command introduced before Junos OS Release 7.4.

`statistics` option introduced in Junos OS Release 16.1 for MX Series routers.

RELATED DOCUMENTATION

show policy damping

test policy

show policy conditions

IN THIS SECTION

Syntax

```
show policy conditions
<condition-name>
<detail>
<dynamic>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show policy conditions
<condition-name>
```

```
<detail>
<dynamic>
```

Description

Display all the configured conditions as well as the routing tables with which the configuration manager is interacting. If the `detail` keyword is included, the output also displays dependent routes for each condition.

Options

none	Display all configured conditions and associated routing tables.
condition-name	(Optional) Display information about the specified condition only.
detail	(Optional) Display the specified level of output.
dynamic	(Optional) Display information about the conditions in the dynamic database.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 30 on page 592 lists the output fields for the **show policy conditions** command. Output fields are listed in the approximate order in which they appear.

Table 30: show policy conditions Output Fields

Field Name	Field Description	Level of Output
Condition	Name of configured condition.	All levels
event	Condition type. If the `if-route-exists` option is configured, the event type is: Existence of a route in a specific routing table.	All levels
Dependent routes	List of routes dependent on the condition, along with the latest generation number.	`detail`
Condition tables	List of routing tables associated with the condition, along with the latest generation number and number of dependencies.	All levels
If-route-exists conditions	List of conditions configured to look for a route in the specified table.	All levels

Sample Output

show policy conditions detail

```
user@host> show policy conditions detail
Configured conditions:
Condition cond1, event: Existence of a route in a specific routing table
Dependent routes:
 172.16.4.4/32, generation 3
 6.6.6.6/32, generation 3
 10.10.10.10/32, generation 3

Condition cond2, event: Existence of a route in a specific routing table
Dependent routes:
None

Condition tables:
```

```
Table inet.0, generation 4, dependencies 3, If-route-exists conditions: cond1 (static) cond2
(static)
```

Release Information

Command introduced in Junos OS Release 9.0.

show ripng general-statistics

IN THIS SECTION

Syntax

```
show ripng general-statistics
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switch)

```
show ripng general-statistics
```

Description

Display general RIP next-generation (RIPng) statistics.

Options

none	Display general RIPng statistics.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 31 on page 594 lists the output fields for the `show ripng general-statistics` command. Output fields are listed in the approximate order in which they appear.

Table 31: show ripng general-statistics Output Fields

Field Name	Field Description
bad msgs	Number of invalid messages received.

Table 31: show ripng general-statistics Output Fields *(Continued)*

Field Name	Field Description
no recv intf	Number of packets received with no matching interface.
curr memory	Amount of memory currently used by RIPng.
max memory	Most memory used by RIPng.

Sample Output

show ripng general-statistics

```
user@host> show ripng general-statistics
RIPng I/O info:
     bad msgs       :         0
     no recv intf   :         0
     curr memory    :         0
     max memory     :         0
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

clear ripng general-statistics | 566

show ripng neighbor

IN THIS SECTION

Syntax

```
show ripng neighbor
<logical-system (all | logical-system-name)>
<name>
```

Syntax (EX Series Switch)

```
show ripng neighbor
<name>
```

Description

Display information about RIP next-generation (RIPng) neighbors.

Options

none	Display information about all RIPng neighbors.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
name	(Optional) Display detailed information about a specific RIPng neighbor.

Required Privilege Level

view

Output Fields

Table 32 on page 597 lists the output fields for the `show ripng neighbor` command. Output fields are listed in the approximate order in which they appear.

Table 32: show ripng neighbor Output Fields

Field Name	Field Description
Neighbor	Name of RIPng neighbor.
State	State of the connection: **Up** or **Dn** (Down).
Source Address	Source address.
Destination Address	Destination address.
Send	Send options: **broadcast**, **multicast**, **none**, **version 1**, or **yes.**
Recv	Type of packets to accept: **both**, **none**, **version 1**, or **yes.**

Table 32: show ripng neighbor Output Fields *(Continued)*

Field Name	Field Description
In Met	Metric added to incoming routes when advertising into RIPng routes that were learned from other protocols.

Sample Output

show ripng neighbor

```
user@host> show ripng neighbor
                    Source                           Dest              In
Neighbor     State  Address                          Address  Send Recv Met
--------     -----  -------                          -------  ---- ---- ---
fe-0/0/2.0      Up  fe80::290:69ff:fe68:b002         ff02::9  yes  yes    1
```

Release Information

Command introduced before Junos OS Release 7.4.

show ripng statistics

IN THIS SECTION

Syntax

```
show ripng statistics
<logical-system (all | logical-system-name)>
<name>
```

Syntax (EX Series Switch)

```
show ripng statistics
<name>
```

Description

Display RIP next generation (RIPng) statistics about messages sent and received on an interface, as well as information received from advertisements from other routing devices.

Options

none	Display RIPng statistics for all neighbors.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

name (Optional) Display detailed information about a specific RIPng neighbor.

Required Privilege Level

view

Output Fields

Table 33 on page 600 lists the output fields for the `show ripng statistics` command. Output fields are listed in the approximate order in which they appear.

Table 33: show ripng statistics Output Fields

Field Name	Field Description
RIPng info	Information about RIPng on the specified interface: • **port**—UDP port number used for RIPng. • **holddown**—Hold-down interval, in seconds. • **rts learned**—Number of routes learned through RIPng. • **rts held down**—Number of routes held down by RIPng. • **rqsts dropped**—Number of received request packets that were dropped. • **resps dropped**—Number of received response packets that were dropped. • **restart**—Graceful restart status. Displayed when RIPng is or has been in the process of graceful restart.

Table 33: show ripng statistics Output Fields *(Continued)*

Field Name	Field Description
logical-interface	Name of the logical interface and its statistics: • **routes learned**—Number of routes learned on the logical interface. • **routes advertised**—Number of routes advertised by the logical interface. • **timeout**—Timeout interval, in seconds. • **update interval**—Interval between routing table updates, in seconds.
Counter	List of counter types: • **Updates Sent**—Number of update messages sent. • **Triggered Updates Sent**—Number of triggered update messages sent. • **Responses Sent**—Number of response messages sent. • **Bad Messages**—Number of invalid messages received. • **Updates Received**—Number of RIPng update messages received. • **Bad Route Entries**—Number of RIPng invalid route entry messages received. • **Updates Ignored**—Number of RIPng update messages ignored. • **RIPng Requests Received**—Number of RIPng request messages received. • **RIPng Requests Ignored**—Number of RIPng request messages ignored.
Total	Total number of packets for the selected counter.
Last 5 min	Number of packets for the selected counter in the most recent 5-minute period.
Last minute	Number of packets for the selected counter in the most recent 1-minute period.

Sample Output

show ripng statistics

```
user@host> show ripng statistics
RIPng info: port 521; holddown 120s;
    rts learned  rts held down  rqsts dropped  resps dropped
              0              0              0              0

so-0/1/3.0:  0 routes learned; 1 routes advertised; timeout 180s; update interval 20s
Counter                       Total   Last 5 min  Last minute
-------                 -----------  -----------  -----------
Updates Sent                    934           16            4
Triggered Updates Sent            1            0            0
Responses Sent                    0            0            0
Bad Messages                      0            0            0
Updates Received                  0            0            0
Bad Route Entries                 0            0            0
Updates Ignored                   0            0            0
RIPng Requests Received           0            0            0
RIPng Requests Ignored            0            0            0
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

clear ripng statistics | 568

show route

IN THIS SECTION

Syntax

```
show route
<all>
<destination-prefix>
<logical-system (all | logical-system-name)>
<private>
<te-ipv4-prefix-ip  te-ipv4-prefix-ip>
<te-ipv4-prefix-node-ip te-ipv4-prefix-node-ip>
<te-ipv4-prefix-node-iso te-ipv4-prefix-node-iso>
<te-ipv6-prefix-ipv6-addr te-ipv6-prefix-ipv6-addr>
<te-ipv6-prefix-node-iso te-ipv6-prefix-node-iso>
<rib-sharding (main | rib-shard-name)>
```

Syntax (EX Series Switches)

```
show route
<all>
<destination-prefix>
<private>
```

Description

Display the active entries in the routing tables.

Options

none	Display brief information about all active entries in the routing tables.
all	(Optional) Display information about all routing tables, including private, or internal, routing tables.
destination-prefix	(Optional) Display active entries for the specified address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
private	(Optional) Display information only about all private, or internal, routing tables.
programmed detail	(Optional) Display API-programmed routes.
display-client-data	(Optional) Display client id and cookie information for routes installed by the routing protocol process client applications.
te-ipv4-prefix-ip *te-ipv4-prefix-ip*	(Optional) Display IPv4 address of the traffic-engineering prefix, without the mask length if present in the routing table.

te-ipv4-prefix-node-ip ***te-ipv4-prefix-node-ip***	(Optional) Display all prefixes that have originated from the traffic-engineering node. You can filter IPv4 node addresses from the traffic-engineered routes in the `lsdist.0` table.
te-ipv4-prefix-node-iso ***te-ipv4-prefix-node-iso***	(Optional) Display all prefixes that have originated from the traffic-engineering node. You can filter IPv4 routes with the specified ISO circuit ID from the `lsdist.0` table.
te-ipv6-prefix-ipv6-addr ***te-ipv6-prefix-ipv6-addr***	(Optional) Filter IPv6 node addresses from the traffic-engineering IPv6 prefix.
te-ipv6-prefix-node-iso ***te-ipv6-prefix-node-iso***	(Optional) Filter IPv6 routes with the specified ISO circuit ID in the traffic-engineering IPv6 prefix.
rib-sharding (main | ***rib-shard-name*****)**	(Optional) Display the rib shard name.

Required Privilege Level

view

Output Fields

Table 34 on page 605 describes the output fields for the `show route` command. Output fields are listed in the approximate order in which they appear.

Table 34: show route Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).
number destinations	Number of destinations for which there are routes in the routing table.

Table 34: show route Output Fields *(Continued)*

Field Name	Field Description
number routes	Number of routes in the routing table and total number of routes in the following states: • `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. 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. If you have configured `uRPF-loose` mode, the holddown bit is most likely set because Kernel Routing Table (KRT) is using inactive route to build valid incoming interfaces. In this case, you can ignore the holddown state because nothing is wrong. • `hidden` (routes that are not used because of a routing policy).

Table 34: show route Output Fields *(Continued)*

Field Name	Field Description

destination-prefix	Route destination (for example:10.0.0.1/24). Sometimes the route information is presented in another format, such as:
	• *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):
	  • *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*]	Protocol from which the route was learned and the preference value for the route. • `+`—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. 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.

Table 34: show route Output Fields *(Continued)*

Field Name	Field Description
weeks:days hours:minutes:seconds	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	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: • `I`—IGP. • `E`—EGP. • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—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. **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.

Table 34: show route Output Fields *(Continued)*

Field Name	Field Description
encapsulated	Extended next-hop encoding capability enabled for the specified BGP community for routing IPv4 traffic over IPv6 tunnels. When BGP receives routes without the tunnel community, IPv4-0ver IPv6 tunnels are not created and BGP routes are resolved without encapsulation.
Route Labels	Stack of labels carried in the BGP route update.
validation-state	(BGP-learned routes) Validation status of the route: • 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. • Unverified—Indicates that the origin of the prefix is not verified against the database. This is because the database got populated and the validation is not called for in the BGP import policy, although origin validation is enabled, or the origin validation is not enabled for the BGP peers. • Valid—Indicates that the prefix and autonomous system pair are found in the database.
to	Next hop to the destination. An angle bracket (>) indicates that the route is the selected route. If the destination is `Discard`, traffic is dropped.

Table 34: show route Output Fields *(Continued)*

Field Name	Field Description
via	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: • 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). For VPNs, expect to see multiple push operations, corresponding to the inner and outer labels required for VPN routes (in the case of a direct PE-to-PE connection, the VPN route would have the inner label push only).
Private unicast	(Enhanced subscriber management for MX Series routers) Indicates that an access-internal route is managed by enhanced subscriber management. By contrast, access-internal routes not managed by enhanced subscriber management are displayed with associated next-hop and media access control (MAC) address information.
balance	Distribution of the load based on the underlying operational interface bandwidth for equal-cost multipaths (ECMP) across the nexthop gateways in percentages.

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

The following sample output shows route hierarchy for translation route.

```
user@host> show route 10.1.1.1

C1.inet.0: 44 destinations, 44 routes (44 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.1.1.1/32        *[PRPD/10] 00:16:50, metric 2
                    >  to 192.0.2.2 via ge-0/0/1.0
```

show route forwarding-table matching 10.1.1.1

```
user@host> show route forwarding-table matching 10.1.1.1
Routing table: C1.inet
Internet:
Destination        Type RtRef Next hop           Type Index    NhRef Netif

10.1.1.1/32        user     0                    indr  1048574     4

                                                 comp      624     2
```

show route 10.1.1.1 extensive expanded-nh

```
user@host> show route 10.1.1.1extensive expanded-nh
C1.inet
C1.inet.0: 44 destinations, 44 routes (44 active, 0 holddown, 0 hidden)
10.1.1.1/32 (1 entry, 1 announced)
Installed-nexthop:
Indr (0xc5c207c) ::44.0.0.1
  Krt_inh (0xc6fd004) Index:1048574 PNH: ::44.0.0.1
    Translate-comp (0xc5c2144) Index:624 v4tov6 src ::22.0.0.1 dest ::44.0.0.1
```

show route te-ipv6-prefix-ipv6-addr

```
user@host> show route te-ipv6-prefix-ipv6-addr 10::10

inet.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)

inet.3: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

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

mpls.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)

inet6.0: 10 destinations, 11 routes (10 active, 0 holddown, 0 hidden)

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

lsdist.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
```

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

IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:10::10/128 } ISIS-L1:0 }/1216
                   *[IS-IS/15] 00:07:58
                       Fictitious
```

show route te-ipv6-prefix-node-iso

```
user@host> show route te-ipv6-prefix-node-iso 0100.0100.0100.00

inet.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)

inet.3: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

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

mpls.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)

inet6.0: 10 destinations, 11 routes (10 active, 0 holddown, 0 hidden)

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

lsdist.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:10::10/128 } ISIS-L1:0 }/
1216
                   *[IS-IS/15] 00:08:46
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:21:0:1::1/128 } ISIS-L1:0 }/
1216
                   *[IS-IS/15] 00:08:46
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:abcd::128:207:200:16/128 } ISIS-
L1:0 }/1216
                   *[IS-IS/15] 00:08:46
                       Fictitious
```

show route (VPN)

The following sample output shows a VPN route with composite next hops enabled. The first `Push` operation corresponds to the outer label. The second `Push` operation corresponds to the inner label.

```
user@host> show route 192.0.2.0

13979:665001.inet.0: 871 destinations, 3556 routes (871 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.0.2.0/24        [BGP/170] 00:28:32, localpref 100, from 10.9.9.160
                      AS path: 13980 ?, validation-state: unverified
                    > to 10.100.0.42 via ae2.0, Push 16, Push 300368(top)
                    [BGP/170] 00:28:28, localpref 100, from 10.9.9.169
                      AS path: 13980 ?, validation-state: unverified
                    > to 10.100.0.42 via ae2.0, Push 126016, Push 300368(top)
                   #[Multipath/255] 00:28:28, metric2 102
                    > to 10.100.0.42 via ae2.0, Push 16, Push 300368(top)
                      to 10.100.0.42 via ae2.0, Push 16, Push 300368(top)
```

show route (with Destination Prefix)

```
user@host> show route 192.168.0.0/12

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

192.168.0.0/12      *[Static/5] 2w4d 12:54:27
                    > to 192.168.167.254 via fxp0.0
```

show route destination-prefix detail

```
user@host> show route 198.51.100.0 detail

inet.0: 15 destinations, 20 routes (15 active, 0 holddown, 0 hidden)
198.51.100.0/24 (2 entries, 2 announced)
        *BGP    Preference: 170/-101
         ...
         BGP-Static Preference: 4294967292
```

```
          Next hop type: Discard
          Address: 0x9041ae4
          Next-hop reference count: 2
          State: <NoReadvrt Int Ext AlwaysFlash>
          Inactive reason: Route Preference
          Local AS:   200
          Age: 4d 1:40:40
          Validation State: unverified
          Task: RT
          Announcement bits (1): 2-BGP_RT_Background
          AS path: 4 5 6 I
```

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 203.0.113.1
           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: 64510 Peer AS: 64511
           Age: 3  Metric2: 1
           Validation State: unverified
           Task: BGP_64510.10.0.0.30+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:64502:100 encapsulation:0L:14
           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 programmed detail

```
user@host> show route programmed detail
inet.0: 36 destinations, 37 routes (36 active, 0 holddown, 0 hidden)
100.75.1.0/27 (2 entries, 1 announced)
        *Static Preference: 5/100
                Next hop type: Router, Next hop index: 0
                Address: 0xcc38a10
                Next-hop reference count: 1
                Next hop: 100.30.1.2 via ge-0/0/2.0 weight 0x1, selected
                Session Id: 0x0
                Next hop: via fti0.1001 weight 0x8001
                Session Id: 0x0
               State: <Active Int NSR-incapable Programmed>
                Age: 37
                Validation State: unverified
                Announcement bits (1): 0-KRT
                AS path: I
```

Release Information

Command introduced before Junos OS Release 7.4.

Option `private` introduced in Junos OS Release 9.5.

Option `private` introduced in Junos OS Release 9.5 for EX Series switches.

Option `display-client-data` introduced in Junos OS Release 16.2R1 on MX80, MX104, MX240, MX480, MX960, MX2010, MX2020, vMX Series routers.

Options `te-ipv4-prefix-ip`, `te-ipv4-prefix-node-ip`, and `te-ipv4-prefix-node-iso` introduced in Junos OS Release 17.2R1 on MX Series and PTX Series.

`rib-sharding` option introduced in cRPD Release 20.1R1.

RELATED DOCUMENTATION

Understanding IS-IS Configuration

Verifying and Managing Junos OS Enhanced Subscriber Management

show route active-path

IN THIS SECTION

Syntax

```
show route active-path
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route active-path
<brief | detail | extensive | terse>
```

Description

Display all active routes for destinations. An active route is a route that is selected as the best path. Inactive routes are not displayed.

Options

none	Display all active routes.
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 | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 active-path

```
user@host> show route active-path

inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

10.255.70.19/32    *[Direct/0] 21:33:52
                    > via lo0.0
10.255.71.50/32    *[IS-IS/15] 00:18:13, metric 10
                    > to 172.16.100.1 via so-2/1/3.0
172.16.100.1/24       *[Direct/0] 00:18:36
                    > via so-2/1/3.0
172.16.100.1/32       *[Local/0] 00:18:41
                      Local via so-2/1/3.0
192.168.64.0/21    *[Direct/0] 21:33:52
                    > via fxp0.0
192.168.70.19/32   *[Local/0] 21:33:52
                      Local via fxp0.0
```

show route active-path brief

The output for the `show route active-path brief` command is identical to that for the `show route active-path` command. For sample output, see `show route active-path`.

show route active-path detail

```
user@host> show route active-path detail

inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)

10.255.70.19/32 (1 entry, 1 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via lo0.0, selected
                State: <Active Int>
                Local AS:   200
```

```
                Age: 21:37:10
                Task: IF
                Announcement bits (3): 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

10.255.71.50/32 (1 entry, 1 announced)
        *IS-IS  Preference: 15
                Level: 1
                Next hop type: Router, Next hop index: 397
                Next-hop reference count: 4
                Next hop: 172.16.100.1 via so-2/1/3.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:31      Metric: 10
                Task: IS-IS
                Announcement bits (4): 0-KRT 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

172.16.100.0/24 (1 entry, 1 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via so-2/1/3.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:54
                Task: IF
                Announcement bits (3): 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

172.16.100.1/32 (1 entry, 1 announced)
        *Local  Preference: 0
                Next hop type: Local
                Next-hop reference count: 11
                Interface: so-2/1/3.0
                State: <Active NoReadvrt Int>
                Local AS:   200
                Age: 21:59
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

192.168.64.0/21 (1 entry, 1 announced)
```

```
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via fxp0.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:37:10
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

192.168.70.19/32 (1 entry, 1 announced)
        *Local  Preference: 0
                Next hop type: Local
                Next-hop reference count: 11
                Interface: fxp0.0
                State: <Active NoReadvrt Int>
                Local AS:   200
                Age: 21:37:10
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I
```

show route active-path extensive

```
user@host> show route active-path extensive

inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)
10.255.70.19/32 (1 entry, 1 announced)
TSI:
IS-IS level 1, LSP fragment 0
IS-IS level 2, LSP fragment 0
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via lo0.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:39:47
                Task: IF
                Announcement bits (3): 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
```

```
                AS path: I

10.255.71.50/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 10.255.71.50/32 -> {172.16.100.1}
IS-IS level 2, LSP fragment 0
        *IS-IS  Preference: 15
                Level: 1
                Next hop type: Router, Next hop index: 397
                Next-hop reference count: 4
                Next hop: 172.16.100.1 via so-2/1/3.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 24:08      Metric: 10
                Task: IS-IS
                Announcement bits (4): 0-KRT 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

172.16.100.1/24 (1 entry, 1 announced)
TSI:
IS-IS level 1, LSP fragment 0
IS-IS level 2, LSP fragment 0
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via so-2/1/3.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 24:31
                Task: IF
                Announcement bits (3): 2-IS-IS 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

172.16.100.1/32 (1 entry, 1 announced)
        *Local  Preference: 0
                Next hop type: Local
                Next-hop reference count: 11
                Interface: so-2/1/3.0
                State: <Active NoReadvrt Int>
                Local AS:   200
                Age: 24:36
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
```

```
                AS path: I

192.168.64.0/21 (1 entry, 1 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 3
                Next hop: via fxp0.0, selected
                State: <Active Int>
                Local AS:   200
                Age: 21:39:47
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I

192.168.70.19/32 (1 entry, 1 announced)
        *Local  Preference: 0
                Next hop type: Local
                Next-hop reference count: 11
                Interface: fxp0.0
                State: <Active NoReadvrt Int>
                Local AS:   200
                Age: 21:39:47
                Task: IF
                Announcement bits (2): 5-Resolve tree 2 6-Resolve tree 3
                AS path: I
```

show route active-path terse

```
user@host> show route active-path terse

inet.0: 7 destinations, 7 routes (6 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 10.255.70.19/32    D   0                       >lo0.0
* 10.255.71.50/32    I  15         10            >172.16.100.1.
* 172.16.100.0/24       D   0                       >so-2/1/3.0
* 172.16.100.2/32       L   0                        Local
* 192.168.64.0/21    D   0                       >fxp0.0
```

```
* 192.168.70.19/32   L    0                        Local
```

Release Information

Command introduced in Junos OS Release 8.0.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route advertising-protocol

IN THIS SECTION

Syntax

```
show route advertising-protocol protocol neighbor-address
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Description

Display the routing information as it has been prepared for advertisement to a particular neighbor of a particular dynamic routing protocol.

Options

brief | detail | extensive | terse	(Optional) Display the specified level of output.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
neighbor-address	Address of the neighboring router to which the route entry is being transmitted.
protocol	Protocol transmitting the route: • `bgp`—Border Gateway Protocol • `dvmrp`—Distance Vector Multicast Routing Protocol • `msdp`—Multicast Source Discovery Protocol • `pim`—Protocol Independent Multicast • `rip`—Routing Information Protocol • `ripng`—Routing Information Protocol next generation

Additional Information

Routes displayed are routes that the routing table has exported into the routing protocol and that have been filtered by the associated protocol's `export` routing policy statements. Starting with Junos OS Release 13.3, you can display the routing instance table `foo` for any address family, on a VPN route reflector, or a VPN AS boundary router that is advertising local VPN routes. However, If you do not specify the `table` in the command, the output displays each VRF prefix twice.

Required Privilege Level

view

Output Fields

Table 35 on page 626 lists the output fields for the `show route advertising-protocol` command. Output fields are listed in the approximate order in which they appear.

Table 35: show route advertising-protocol Output Fields

Field Name	Field Description	Level of Output
routing-table-name	Name of the routing table—for example, inet.0.	All levels
number destinations	Number of destinations for which there are routes in the routing table.	All levels
number routes	Number of routes in the routing table and total number of routes in the following states: • `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)	All levels

Table 35: show route advertising-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Prefix	Destination prefix.	`brief` none
destination-prefix (entry , announced)	Destination prefix. 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.	`detail extensive`
BGP group and type	BGP group name and type (Internal or External).	`detail extensive`
Route Distinguisher	Unique 64-bit prefix augmenting each IP subnet.	`detail extensive`
Advertised Label	Incoming label advertised by the Label Distribution Protocol (LDP). When an IP packet enters a label-switched path (LSP), the ingress router examines the packet and assigns it a label based on its destination, placing the label in the packet's header. The label transforms the packet from one that is forwarded based on its IP routing information to one that is forwarded based on information associated with the label.	`detail extensive`
Label-Base, range	First label in a block of labels and label block size. A remote PE router uses this first label when sending traffic toward the advertising PE router.	`detail extensive`
VPN Label	Virtual private network (VPN) label. Packets are sent between CE and PE routers by advertising VPN labels. VPN labels transit over either a Resource Reservation Protocol (RSVP) or a Label Distribution Protocol (LDP) label-switched path (LSP) tunnel.	`detail extensive`
Nexthop	Next hop to the destination. An angle bracket (>) indicates that the route is the selected route. If the next-hop advertisement to the peer is Self, and the RIB-out next hop is a specific IP address, the RIB-out IP address is included in the extensive output.	All levels

Table 35: show route advertising-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
MED	Multiple exit discriminator value included in the route.	`brief`
Lclpref or Localpref	Local preference value included in the route.	All levels
Queued	When BGP route prioritization is enabled and a route is present in a priority queue, this shows which priority queue the route is in.	All levels except `brief`
AS path	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: • `I`—IGP. • `E`—EGP. • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—Brackets enclose the local AS number associated with the AS path if configured on the router, 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. **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.	All levels
Route Labels	Stack of labels carried in the BGP route update.	`detail extensive`

Table 35: show route advertising-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Cluster list	(For route reflected output only) Cluster ID sent by the route reflector.	`detail` `extensive`
Originator ID	(For route reflected output only) Address of routing device that originally sent the route to the route reflector.	`detail` `extensive`
`Communities`	Community path attribute for the route. See the output field table for the *show route detail* command for all possible values for this field.	`detail` `extensive`
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.	`detail` `extensive`
Attrset AS	Number, local preference, and path of the autonomous system (AS) that originated the route. These values are stored in the Attrset attribute at the originating router.	`detail` `extensive`
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).	`detail` `extensive`
control flags	Control flags: `none` or `Site Down`.	`detail` `extensive`
mtu	Maximum transmission unit (MTU) of the Layer 2 circuit.	`detail` `extensive`

Sample Output

show route advertising-protocol bgp (Layer 3 VPN)

```
user@host> show route advertising-protocol bgp 10.255.14.171
      VPN-A.inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
      Prefix            Nexthop              MED    Lclpref AS path
      10.255.14.172/32  Self                   1        100 I
      VPN-B.inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
```

```
      Prefix              Nexthop               MED     Lclpref AS path
      10.255.14.181/32    Self                    2        100 I
```

show route advertising-protocol bgp (IPv6 unicast address community)

```
user@host> show route advertising-protocol bgp 10.255.165.220 extensive
       vpn1.inet.0: 13 destinations, 13 routes (13 active, 0 holddown, 0 hidden)
    * 128.205.172.129/32 (1 entry, 1 announced)
    BGP group internal type Internal
    Route Distinguisher: 10.255.168.42:8
    VPN Label: 299808
    Nexthop: Self
    Flags: Nexthop Change
    MED: 1
    Localpref: 100
       AS path: [203] I
       Communities: rte-type:0.0.0.0:1:0 ipv6-origin:<2001:db8:9999::9>:89
       ipv6-target:<2001:db8:9999::9>:89 ipv6-extended:203:<2001:db8:9999::9>:89
       ipv6-origin:<2001:db8:9999::9>:137 ipv6-target:<2001:db8:9999::9>:137
       ipv6-extended:515:<2001:db8:9999::9>:137
```

show route advertising-protocol bgp detail

```
user@host> show route advertising-protocol bgp 10.222.1.3 detail
bgp20.inet.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
10.222.1.11/32 (1 entry, 1 announced)
 BGP group pe-pe type Internal
     Route Distinguisher: 10.255.14.11:69
     Advertised Label: 100000
     next hop: Self
     Localpref: 100
     AS path: 2 I
     Communities: target:69:20
     AIGP 210
10.8.0.0/16 (1 entry, 1 announced)
 BGP group pe-pe type Internal
     Route Distinguisher: 10.255.14.11:69
     Advertised Label: 100000
     Next hop: Self
     Localpref: 100
```

```
     AS path: 2 I
     Communities: target:69:20
     AIGP 210
```

show route advertising-protocol bgp detail (Aggregate Extended Community Bandwidth)

```
user@host> show route advertising-protocol bgp 10.0.4.2 10.0.2.0/30 detail
inet.0: 20 destinations, 26 routes (20 active, 0 holddown, 0 hidden)
* 10.0.2.0/30 (2 entries, 1 announced)
 BGP group external2 type External
     Nexthop: Self
     AS path: [65000] 65001 I
     Communities: bandwidth:65000:80000000
```

show route advertising-protocol bgp detail (BGP Multicast)

```
user@host>show route advertising-protocol bgp 10.4.6.1 detail
bgpmcast.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
* 4:3:0:0:32:10.2.20.20:32:225.1.1.1:10.1.1.1:10.20.20.20/240 (1 entry, 1 announced)
 BGP group ibgp type Internal
     Nexthop: 10.4.6.6
     Localpref: 100
     AS path: [65000] 65001 I
     Communities: target:65000:100
     Tunnel type: AnyEncap, RPF tunnel:, Remote end point: 10.1.2.1
     Tunnel type: AnyEncap, Remote end point: 10.1.4.1
```

show route advertising-protocol bgp detail (Labeled Unicast)

```
user@host>show route advertising bgp 10.1.1.3 detail
inet.0: 69 destinations, 70 routes (69 active, 0 holddown, 0 hidden)
* 10.1.1.8/32 (2 entries, 2 announced)
BGP group ibgp type Internal
Route Labels: 1000123(top) 1000124 1000125 1000126
Nexthop: 10.1.1.4
MED: 7
Localpref: 100
AS path: [65005] I
```

```
Cluster ID: 10.3.3.3
Originator ID: 10.1.1.1
Entropy label capable
inet6.0: 26 destinations, 28 routes (26 active, 0 holddown, 0 hidden)
* 2001:db8:100::1/128 (2 entries, 1 announced)
BGP group ibgp type Internal
Labels: 1000123(top) 1000124 1000125 1000126
Nexthop: 2001:db8:0:ffff:10.1.1.4
Localpref: 100
AS path: [65005] I
Cluster ID: 10.3.3.3
Originator ID: 10.1.1.1
```

show route advertising-protocol bgp detail (Layer 2 VPN)

```
user@host> show route advertising-protocol bgp 192.168.24.1 detail
vpn-a.l2vpn.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
192.168.16.10:1:1:1/96 (1 entry, 1 announced)
 BGP group int type Internal
     Route Distinguisher: 192.168.16.1:1
     Label-base : 32768, range : 3
     Nexthop: Self
     Localpref: 100
     AS path: I
     Communities: target:65412:100
     AIGP 210
     Layer2-info: encaps:VLAN, control flags:, mtu:
```

show route advertising-protocol bgp detail (Layer 3 VPN)

```
user@host> show route advertising-protocol bgp 10.255.14.176 detail
vpna.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
* 10.49.0.0/30 (1 entry, 1 announced)
 BGP group ibgp type Internal
     Route Distinguisher: 10.255.14.174:2
     VPN Label: 101264
     Nexthop: Self
     Localpref: 100
     AS path: I
     Communities: target:10:100
```

```
     AIGP 210
     AttrSet AS: 65100
         Localpref: 100
         AS path: I
...
```

show route advertising-protocol bgp extensive all (Next Hop Self with RIB-out IP Address)

```
user@host> show route advertising-protocol bgp 10.200.0.2 10.170.1.0/24 extensive all
inet.0: 13 destinations, 19 routes (13 active, 0 holddown, 6 hidden)
  10.170.1.0/24 (2 entries, 1 announced)
 BGP group eBGP-INTEROP type External
     Nexthop: Self (rib-out 10.100.3.2)
     AS path: [64713] 65200 I
...
```

Release Information

Command introduced before Junos OS Release 7.4.

`ipv6-origin`, `ipv6-target`, and `ipv6-extended Community` output fields supported in Junos OS Release and Junos OS Evolved Release 23.1.

RELATED DOCUMENTATION

Example: Configuring the MED Attribute That Determines the Exit Point in an AS

show route all

IN THIS SECTION

Syntax

```
show route all
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route all
```

Description

Display information about all routes in all routing tables, including private, or internal, tables.

Options

none	Display information about all routes in all routing tables, including private, or internal, tables.

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

Required Privilege Level

view

Output Fields

In Junos OS Release 9.5 and later, only the output fields for the `show route all` command display all routing tables, including private, or hidden, routing tables. The output field table of the *show route* command does not display entries for private, or hidden, routing tables in Junos OS Release 9.5 and later.

Sample Output

show route all

The following example displays a snippet of output from the `show route` command and then displays the same snippet of output from the `show route all` command:

```
user@host> show route
mpls.0: 7 destinations, 7 routes (5 active, 0 holddown, 2 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
0                  *[MPLS/0] 2d 02:24:39, metric 1
                      Receive
1                  *[MPLS/0] 2d 02:24:39, metric 1
                      Receive
2                  *[MPLS/0] 2d 02:24:39, metric 1
                      Receive
800017             *[VPLS/7] 1d 14:00:16
                    > via vt-3/2/0.32769, Pop
800018             *[VPLS/7] 1d 14:00:26
                    > via vt-3/2/0.32772, Pop
```

```
user@host> show route all
mpls.0: 7 destinations, 7 routes (5 active, 0 holddown, 2 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
0                  *[MPLS/0] 2d 02:19:12, metric 1
                      Receive
1                  *[MPLS/0] 2d 02:19:12, metric 1
                      Receive
2                  *[MPLS/0] 2d 02:19:12, metric 1
                      Receive
800017             *[VPLS/7] 1d 13:54:49
                    > via vt-3/2/0.32769, Pop
800018             *[VPLS/7] 1d 13:54:59
                    > via vt-3/2/0.32772, Pop
vt-3/2/0.32769      [VPLS/7] 1d 13:54:49
                      Unusable
vt-3/2/0.32772      [VPLS/7] 1d 13:54:59
                      Unusable
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route brief

show route detail

show route best

IN THIS SECTION

Syntax

```
show route best destination-prefix
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route best destination-prefix
<brief | detail | extensive | terse>
```

Description

Display the route in the routing table that is the best route to the specified address or range of addresses. The best route is the longest matching route.

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`.
destination-prefix	Address or range of addresses.
logical-system (all | ***logical-system-name*****)**	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 best

```
user@host> show route best 10.255.70.103
inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
10.255.70.103/32   *[OSPF/10] 1d 13:19:20, metric 2
                    > to 10.31.1.6 via ge-3/1/0.0
                      via so-0/3/0.0

inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
```

```
10.255.70.103/32   *[RSVP/7] 1d 13:20:13, metric 2
                    > via so-0/3/0.0, label-switched-path green-r1-r3

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
10.0.0.0/8         *[Direct/0] 2d 01:43:34
                    > via fxp2.0
                    [Direct/0] 2d 01:43:34
                    > via fxp1.0
```

show route best detail

```
user@host> show route best 10.255.70.103 detail
inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
10.255.70.103/32 (1 entry, 1 announced)
        *OSPF   Preference: 10
                Next-hop reference count: 9
                Next hop: 10.31.1.6 via ge-3/1/0.0, selected
                Next hop: via so-0/3/0.0
                State: <Active Int>
                Local AS:    69
                Age: 1d 13:20:06        Metric: 2
                Area: 0.0.0.0
                Task: OSPF
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

inet.3: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
Restart Complete
10.255.70.103/32 (1 entry, 1 announced)
        State: <FlashAll>
        *RSVP   Preference: 7
                Next-hop reference count: 5
                Next hop: via so-0/3/0.0 weight 0x1, selected
                Label-switched-path green-r1-r3
                Label operation: Push 100016
                State: <Active Int>
                Local AS:    69
                Age: 1d 13:20:59        Metric: 2
                Task: RSVP
```

```
                Announcement bits (1): 1-Resolve tree 2
                AS path: I

private1__inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
10.0.0.0/8 (2 entries, 0 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 1
                Next hop: via fxp2.0, selected
                State: <Active Int>
                Age: 2d 1:44:20
                Task: IF
                AS path: I
         Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 1
                Next hop: via fxp1.0, selected
                State: <NotBest Int>
                Inactive reason: No difference
                Age: 2d 1:44:20
                Task: IF
                AS path: I
```

show route best extensive

The output for the `show route best extensive` command is identical to that for the `show route best detail` command. For sample output, see "show route best detail" on page 639.

show route best terse

```
user@host> show route best 10.255.70.103 terse
inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 10.255.70.103/32   O  10          2            >10.31.1.6
                                                  so-0/3/0.0

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

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

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 10.255.70.103/32   R   7          2            >so-0/3/0.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 10.0.0.0/8         D   0                       >fxp2.0
                     D   0                       >fxp1.0
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route brief

show route detail

show route extensive

show route terse

show route brief

IN THIS SECTION

Syntax

```
show route brief
<destination-prefix>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route brief
<destination-prefix>
```

Description

Display brief information about the active entries in the routing tables.

Options

none	Display all active entries in the routing table.
destination-prefix	(Optional) Display active entries for the specified address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

For information about output fields, see the Output Field table of the `show route` command.

Sample Output

show route brief

```
user@host> show route brief
inet.0: 10 destinations, 10 routes (9 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[Static/5] 1w5d 20:30:29
                      Discard
10.255.245.51/32   *[Direct/0] 2w4d 13:11:14
                    > via lo0.0
172.16.0.0/12      *[Static/5] 2w4d 13:11:14
                    > to 192.168.167.254 via fxp0.0
192.168.0.0/18         *[Static/5] 1w5d 20:30:29
                    > to 192.168.167.254 via fxp0.0
192.168.40.0/22    *[Static/5] 2w4d 13:11:14
                    > to 192.168.167.254 via fxp0.0
192.168.64.0/18    *[Static/5] 2w4d 13:11:14
                    > to 192.168.167.254 via fxp0.0
192.168.164.0/22   *[Direct/0] 2w4d 13:11:14
                    > via fxp0.0
192.168.164.51/32  *[Local/0] 2w4d 13:11:14
                      Local via fxp0.0
207.17.136.192/32  *[Static/5] 2w4d 13:11:14
                    > to 192.168.167.254 via fxp0.0
green.inet.0: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
100.101.0.0/16     *[Direct/0] 1w5d 20:30:28
```

```
                     > via fe-0/0/3.0
100.101.2.3/32     *[Local/0] 1w5d 20:30:28
                      Local via fe-0/0/3.0
172.16.233.5/32       *[OSPF/10] 1w5d 20:30:29, metric 1
                      MultiRecv
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route all

show route best

show route detail

IN THIS SECTION

Syntax

```
show route detail
<destination-prefix>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route detail
<destination-prefix>
```

Description

Display detailed information about the active entries in the routing tables.

Options

none	Display all active entries in the routing table on all systems.
destination-prefix	(Optional) Display active entries for the specified address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 36 on page 646 describes the output fields for the `show route detail` command. Output fields are listed in the approximate order in which they appear.

Table 37 on page 656 describes all possible values for the Next-hop Types output field.

Table 38 on page 658 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 39 on page 661 describes the possible values for the Communities output field.

Table 36: show route detail Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).
number destinations	Number of destinations for which there are routes in the routing table.
number routes	Number of routes in the routing table and total number of routes in the following states: • `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 36: show route detail Output Fields *(Continued)*

Field Name	Field Description

route-destination (entry, announced)	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:
	• *`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).
	  ◦ *`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`.
	  ◦ *`source`*—Source of the advertisement: `Local` or `Remote`.

label stacking	(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. • `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).

Table 36: show route detail Output Fields *(Continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th></tr>
<tr><td>[protocol, preference]</td><td>
Protocol from which the route was learned and the preference value for the route.

<code>+</code>—A plus sign indicates the active route, which is the route installed from the routing table into the forwarding table.
<code>-</code> —A hyphen indicates the last active route.
<code>*</code>—An asterisk indicates that the route is both the active and the last active route. An asterisk before a <code>to</code> line indicates the best subpath to the route.

In every routing metric except for the BGP <code>LocalPref</code> attribute, a lesser value is preferred. In order to use common comparison routines, Junos OS stores the 1's complement of the <code>LocalPref</code> value in the <code>Preference2</code> field. For example, if the <code>LocalPref</code> value for Route 1 is 100, the <code>Preference2</code> value is -101. If the <code>LocalPref</code> value for Route 2 is 155, the <code>Preference2</code> value is -156. Route 2 is preferred because it has a higher <code>LocalPref</code> value.
<code>Preference2</code> values are signed integers, that is, <code>Preference2</code> values can be either positive or negative values. However, Junos OS evaluates <code>Preference2</code> values as unsigned integers that are represented by positive values. Based on the <code>Preference2</code> values, Junos OS evaluates a preferred route differently in the following scenarios:

Both Signed <code>Preference2</code> values

Route A = -101
Route B = -156

Where both the <code>Preference2</code> values are signed, Junos OS evaluates only the unsigned value of <code>Preference2</code> and Route A, which has a lower <code>Preference2</code> value is preferred.
Unsigned <code>Preference2</code> values
Now consider both unsigned <code>Preference2</code> values:

Route A = 4294967096
Route B = 200

Here, Junos OS considers the lesser <code>Preference2</code> value and Route B with a <code>Preference2</code> value of 200 is preferred because it is less than 4294967096.
Combination of signed and unsigned <code>Preference2</code> values

</td></tr>
</table>

Table 36: show route detail Output Fields *(Continued)*

Field Name	Field Description
	When `Preference2` values of two routes are compared, and for one route the `Preference2` is a signed value, and for the other route it is an unsigned value, Junos OS prefers the route with the positive `Preference2` value over the negative `Preference2` value. For example, consider the following signed and unsigned `Preference2` values: • Route A = -200 • Route B = 200 In this case, Route B with a `Preference2` value of 200 is preferred although this value is greater than -200, because Junos OS evaluates only the unsigned value of the `Preference2` value.
Level	(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.
Route Distinguisher	IP subnet augmented with a 64-bit prefix.
PMSI	Provider multicast service interface (MVPN routing table).
Next-hop type	Type of next hop. For a description of possible values for this field, see Table 37 on page 656.
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.

Table 36: show route detail Output Fields *(Continued)*

Field Name	Field Description
Next hop	Network layer address of the directly reachable neighboring system.
via	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: • 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 38 on page 658.
Local AS	AS number of the local routing device.

Table 36: show route detail Output Fields *(Continued)*

Field Name	Field Description
Age	How long the route has been known.
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.
Metric*n*	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.
MED-plus-IGP	Metric value for BGP path selection to which the IGP cost to the next-hop destination has been added.
TTL-Action	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. For sample output, see `show route table` .
Task	Name of the protocol that has added the route.
Announcement bits	The number of BGP peers or protocols to which Junos OS has announced this route, followed by the list of the recipients of the announcement. Junos OS can also announce the route to the KRT for installing the route into the Packet Forwarding Engine, to a resolve tree, a L2 VC, or even a VPN. For example, *n*-`Resolve inet` indicates that the specified route is used for route resolution for next hops found in the routing table. • *n*—An index used by Juniper Networks customer support only.

Table 36: show route detail Output Fields *(Continued)*

Field Name	Field Description
AS path	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: • `I`—IGP. • `E`—EGP. • `Recorded`—The AS path is recorded by the sample process (sampled). • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—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. **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.

Table 36: show route detail Output Fields *(Continued)*

Field Name	Field Description
validation-state	(BGP-learned routes) Validation status of the route: • `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. • `Unverified`—Indicates that the origin of the prefix is not verified against the database. This is because the database got populated and the validation is not called for in the BGP import policy, although origin validation is enabled, or the origin validation is not enabled for the BGP peers. • `Valid`—Indicates that the prefix and autonomous system pair are found in the database.
ORR Generation-ID	Displays the optimal route reflection (ORR) generation identifier. ISIS and OSPF interior gateway protocol (IGP) updates filed whenever any of the corresponding ORR route has its metric valued changed, or if the ORR route is added or deleted.
FECs bound to route	Point-to-multipoint root address, multicast source address, and multicast group address when multipoint LDP (M-LDP) inband signaling is configured.
Primary Upstream	When multipoint LDP with multicast-only fast reroute (MoFRR) is configured, the primary upstream path. MoFRR transmits a multicast join message from a receiver toward a source on a primary path, while also transmitting a secondary multicast join message from the receiver toward the source on a backup path.
RPF Nexthops	When multipoint LDP with MoFRR is configured, the reverse-path forwarding (RPF) next-hop information. Data packets are received from both the primary path and the secondary paths. The redundant packets are discarded at topology merge points due to the RPF checks.
Label	Multiple MPLS labels are used to control MoFRR stream selection. Each label represents a separate route, but each references the same interface list check. Only the primary label is forwarded while all others are dropped. Multiple interfaces can receive packets using the same label.

Table 36: show route detail Output Fields *(Continued)*

Field Name	Field Description
weight	Value used to distinguish MoFRR primary and backup routes. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible.
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 39 on page 661 for all possible values for this field.
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).
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 LongLivedStale	The LongLivedStale flag indicates that the route was marked LLGR-stale by this router, as part of the operation of LLGR receiver mode. Either this flag or the LongLivedStaleImport flag may be displayed for a route. Neither of these flags are displayed at the same time as the Stale (ordinary GR stale) flag.

Table 36: show route detail Output Fields *(Continued)*

Field Name	Field Description
Accepted LongLivedStaleImpor t	The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy. Either this flag or the LongLivedStale flag may be displayed for a route. Neither of these flags are displayed at the same time as the Stale (ordinary GR stale) flag. Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and import into the inet.0 routing table
ImportAccepted LongLivedStaleImpor t DeletePending	Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and imported into the inet.0 routing table The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy. The DeletePending flag indicates that a BGP route needs to be processed due to a BGP peer down event.
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.
Statistics ID Group	Indicates the Kernel ID number and Statistics ID number.
Statistics	Indicates the number of packets and data transferred.

Table 37: Next-hop Types Output Field Values

Next-Hop Type	Description
Broadcast (bcast)	Broadcast next hop.
Deny	Deny next hop.
Discard	Discard next hop.
Dynamic List	Dynamic list 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.
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.

Table 37: Next-hop Types Output Field Values *(Continued)*

Next-Hop Type	Description
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	A specific node or set of nodes to which the routing device forwards packets that match the route prefix. To qualify as next-hop type router, the route must meet the following criteria: • 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.
Software	Next hop added to the Routing Engine forwarding table for remote IP addresses with prefix /32 for Junos OS Evolved only.
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 38: 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.

Table 38: State Output Field Values *(Continued)*

Value	Description
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).
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.

Table 38: State Output Field Values *(Continued)*

Value	Description
NSR-incapable	Route added by non-NSR supported protocols.
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.
Programmed	Route installed programatically by on-box or off-box applications using API.
ProtectionCand	Indicates paths requesting protection.
ProtectionPath	Indicates the route entry that can be used as a protection path.
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.

Table 38: State Output Field Values *(Continued)*

Value	Description
Unusable path	Path is not usable because of one of the following conditions: • 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 39: Communities Output Field Values

Value	Description
area-number	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.
bandwidth: *local AS number:link-bandwidth-number*	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.
domain-id	Unique configurable number that identifies the OSPF domain.
domain-id-vendor	Unique configurable number that further identifies the OSPF domain.
link-bandwidth-number	Link-bandwidth number: from 0 through 4,294,967,295 (bytes per second).
local AS number	Local AS number: from 1 through 65,535.
options	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.

Table 39: Communities Output Field Values *(Continued)*

Value	Description
origin	(Used with VPNs) Identifies where the route came from.
ospf-route-type	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 be0); 7 for NSSA routes; or 129 for sham link endpoint addresses.
route-type-vendor	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*.
rte-type	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*.
target	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.
ipv6-origin	Defines the source of the IPv6 unicast address in a policy match condition.
ipv6-target	Defines the VPN IPv6 target unicast address used in a policy match condition. The target has the *128-bit IP address:16-bit number* format. For example, 2001:db8:9999::9.
ipv6-extended	Defines the extended format of the IPv6 unicast address in a policy match. For example, `ipv6-extended:203:<2001:db8::7>:67`
unknown IANA	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.
unknown OSPF vendor community	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:    65069
                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:    65069
                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:    65069
                Age: 1:30:17    Metric: 1
                ORR Generation-ID: 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:    65069
                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:    65069
                Age: 1:29:56    Metric: 2
                Area: 0.0.0.0
                ORR Generation-ID: 1
                        Task: OSPF
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

...

172.16.233.2/32 (1 entry, 1 announced)
        *PIM    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    65069
                Age: 1:31:45
                Task: PIM Recv
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

...

172.16.233.22/32 (1 entry, 1 announced)
```

```
        *IGMP   Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    65069
                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:    65069
                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:    65069
                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:    65069
                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:    65069
                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)

  299840 (1 entry, 1 announced)
  TSI:
  KRT in-kernel 299840 /52 -> {indirect(1048575)}
          *RSVP   Preference: 7/2
                  Next hop type: Flood
                  Address: 0x9174a30
                  Next-hop reference count: 4
                  Next hop type: Router, Next hop index: 798
                  Address: 0x9174c28
                  Next-hop reference count: 2
                  Next hop: 172.16.0.2 via lt-1/2/0.9 weight 0x1
                  Label-switched-path R2-to-R4-2p2mp
                  Label operation: Pop
                  Next hop type: Router, Next hop index: 1048574
                  Address: 0x92544f0
                  Next-hop reference count: 2
                  Next hop: 172.16.0.2 via lt-1/2/0.7 weight 0x1
```

```
                  Label-switched-path R2-to-R200-p2mp
                  Label operation: Pop
                  Next hop: 172.16.0.2 via lt-1/2/0.5 weight 0x8001
                  Label operation: Pop
                  State: <Active Int>
                  Age: 1:29       Metric: 1
                  Task: RSVP
                  Announcement bits (1): 0-KRT
                  AS path: I...

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:    65069
                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:    65069
                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:    65069
                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:    65069
                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:    65069
                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:    65069 Peer AS:    65069
                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

...

l2circuit.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:    65099
                Age: 10:21
                Task: l2 circuit
                Announcement bits (1): 0-LDP
                AS path: I
                VC Label 100000, MTU 1500, VLAN ID 512

inet.0: 45 destinations, 47 routes (44 active, 0 holddown, 1 hidden)
```

```
10.1.1.3/32 (1 entry, 1 announced)
        *IS-IS  Preference: 18
                Level: 2
                Next hop type: Router, Next hop index: 580
                Address: 0x9db6ed0
                Next-hop reference count: 8
                Next hop: 10.1.1.6 via lt-1/0/10.5, selected
                Session Id: 0x18a
                State: <Active Int>
                Local AS:     2
                Age: 1:32       Metric: 10
                Validation State: unverified
                ORR Generation-ID: 1
                Task: IS-IS
                Announcement bits (3): 0-KRT 5-Resolve tree 4 6-Resolve_IGP_FRR task
                AS path: I

inet.0: 61 destinations, 77 routes (61 active, 1 holddown, 0 hidden)
10.1.1.1/32 (2 entries, 1 announced)
        *OSPF   Preference: 10
                Next hop type: Router, Next hop index: 673
                Address: 0xc008830
                Next-hop reference count: 3
                Next hop: 10.1.1.8 via ge-0/0/2.0, selected
                Session Id: 0x1b7
                State: <Active Int>
                Local AS:     1
                Age: 3:06:59    Metric: 100
                Validation State: unverified
                ORR Generation-ID: 1
                Area: 0.0.0.0
                Task: OSPF
                Announcement bits (2): 1-KRT 9-Resolve tree 2
                AS path: I
```

show route programmed detail (with statistics information)

```
user@host> show route programmed detail

inet.0: 104 destinations, 104 routes (103 active, 0 holddown, 1 hidden)
```

```
inet.3: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
192.0.2.0/24 (1 entry, 1 announced)
        *Static Preference: 5/100
                Next hop type: Router, Next hop index: 0
                Address: 0xa667694
                Next-hop reference count: 15
                Next hop: 198.51.100.0 via ae0.0
                Label operation: Push 50008, Push 3400(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3400: None;
                Label element ptr: 0x84e7e90
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 10, Stats IDs = { 536870923 }
                Statistics: Packets 1380, Bytes 681720
                Next hop: 198.51.100.1 via ae0.1
                Label operation: Push 50008, Push 3401(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3401: None;
                Label element ptr: 0x84e7ee0
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 11, Stats IDs = { 536870924 }
                Statistics: Packets 1444, Bytes 713336
                Next hop: 198.51.100.2 via ae0.2
                Label operation: Push 50008, Push 3410(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3410: None;
                Label element ptr: 0xa1926a0
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 12, Stats IDs = { 536870925 }
                Statistics: Packets 1420, Bytes 701480
                Next hop: 198.51.100.3 via ge-0/0/2.0
```

```
                  Label operation: Push 50008, Push 3411(top)
                  Label TTL action: prop-ttl, prop-ttl(top)
                  Load balance label: Label 50008: None; Label 3411: None;
                  Label element ptr: 0xa1926c8
                  Label parent element ptr: 0x0
                  Label element references: 3
                  Label element child references: 0
                  Label element lsp id: 0
                  Session Id: 0x0
                  Statistics ID Group: Kernel ID = 13, Stats IDs = { 536870926 }
                  Statistics: Packets 1486, Bytes 734084
                  Next hop: 198.51.100.4 via ge-0/0/2.1, selected
                  Label operation: Push 50008, Push 3420(top)
                  Label TTL action: prop-ttl, prop-ttl(top)
                  Load balance label: Label 50008: None; Label 3420: None;
                  Label element ptr: 0xa1926f0
                  Label parent element ptr: 0x0
                  Label element references: 3
                  Label element child references: 0
                  Label element lsp id: 0
                  Session Id: 0x0
                  Statistics ID Group: Kernel ID = 14, Stats IDs = { 536870927 }
                  Statistics: Packets 1399, Bytes 691106
                  Next hop: 198.51.100.5 via ge-0/0/2.2
                  Label operation: Push 50008, Push 3421(top)
                  Label TTL action: prop-ttl, prop-ttl(top)
                  Load balance label: Label 50008: None; Label 3421: None;
                  Label element ptr: 0xa192718
                  Label parent element ptr: 0x0
                  Label element references: 3
                  Label element child references: 0
                  Label element lsp id: 0
                  Session Id: 0x0
                  Statistics ID Group: Kernel ID = 15, Stats IDs = { 536870928 }
                  Statistics: Packets 1455, Bytes 718770
                  Next hop: 198.51.100.6 via ge-0/0/4.0
                  Label operation: Push 50008, Push 3450(top)
                  Label TTL action: prop-ttl, prop-ttl(top)
                  Load balance label: Label 50008: None; Label 3450: None;
                  Label element ptr: 0xa192740
                  Label parent element ptr: 0x0
                  Label element references: 3
                  Label element child references: 0
```

```
                 Label element lsp id: 0
                 Session Id: 0x0
                 Statistics ID Group: Kernel ID = 16, Stats IDs = { 536870929 }
                 Statistics: Packets 1407, Bytes 695058
                 Next hop: 198.51.100.7 via ge-0/0/4.1
                 Label operation: Push 50008, Push 3460(top)
                 Label TTL action: prop-ttl, prop-ttl(top)
                 Load balance label: Label 50008: None; Label 3460: None;
                 Label element ptr: 0xa192768
                 Label parent element ptr: 0x0
                 Label element references: 3
                 Label element child references: 0
                 Label element lsp id: 0
                 Session Id: 0x0
                 Statistics ID Group: Kernel ID = 17, Stats IDs = { 536870930 }
                 Statistics: Packets 1418, Bytes 700492
                 Next hop: 198.51.100.8 via ge-0/0/4.2
                 Label operation: Push 50008, Push 3470(top)
                 Label TTL action: prop-ttl, prop-ttl(top)
                 Load balance label: Label 50008: None; Label 3470: None;
                 Label element ptr: 0xa192790
                 Label parent element ptr: 0x0
                 Label element references: 3
                 Label element child references: 0
                 Label element lsp id: 0
                 Session Id: 0x0
                 Statistics ID Group: Kernel ID = 18, Stats IDs = { 536870931 }
                 Statistics: Packets 0, Bytes 0
                 State: <Active Int NSR-incapable Programmed>
                 Age: 12:46
                 Validation State: unverified
                 Announcement bits (3): 0-Resolve tree 6 1-Resolve tree 1 3-Resolve_IGP_FRR task
                 AS path: I
                 Session-IDs associated:
                 Session-id: 370 Version: 0
                 Thread: junos-main
```

show route detail (with IPv6 unicast address community)

```
user@host> show route 203.0.112.1 detail
inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)
```

```
203.0.112.1/32 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 559
                Address: 0x7b5de84
                Next-hop reference count: 12, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 13.1.1.1
                Next hop: 13.1.1.1 via ge-0/0/0.0, selected
                Session Id: 320
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 8:41       Metric: 1
                Validation State: unverified
                Task: BGP_100.13.1.1.1
                Announcement bits (1): 0-KRT
                AS path: 100 I
                Communities: target:100:65 target:172.16.7.7:2 ipv6-target:<2001:db8::5>:65
                ipv6-target:<2001:db8::5>:101 ipv6-origin:<2001:db8::6>:66
                ipv6-origin:<2001:db8::6>:102 ipv6-extended:100:<2001:db8::7>:67
                ipv6-extended:256:<2001:db8::7>:103
                Accepted
                Localpref: 100
                Router ID: 2.2.2.2
                Thread: junos-main
```

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
                  Validation State: unverified
                Task: BGP_2.10.1.1.2+59955
```

```
                Announcement bits (1): 0-KRT
                AS path: 65002 I
                Accepted Multipath
                Localpref: 100
                Router ID: 172.16.1.2
         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:     65001 Peer AS:     65002
                Age: 5:04:43
                   Validation State: unverified
                Task: BGP_2.10.1.1.6+58198
                AS path: 65002 I
                Accepted MultipathContrib
                Localpref: 100
                Router ID: 172.16.1.3
```

show route detail (with BGP, DeletePending)

```
user@host> show route detail
2001:db8:2:1:10.1.1.12/30 (1 entry, 0 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 65002:1
                Next hop type: Indirect
                Address: 0x95c4ee8
                Next-hop reference count: 6
                Source: 10.1.1.4
                Next hop type: Router, Next hop index: 809
                Next hop: 10.1.1.6 via lt-1/0/10.5, selected
                Label operation: Push 299888, Push 299792(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 299888: None; Label 299792: None;
                Session Id: 0x142
                Protocol next hop: 10.1.1.4
                Label operation: Push 299888
                Label TTL action: prop-ttl
```

```
                Load balance label: Label 299888: None;
                Indirect next hop: 0x96f0110 1048574 INH Session ID: 0x14e
                State: <Active Int Ext ProtectionPath ProtectionCand>
                Local AS:     65002 Peer AS:     65002
                Age: 2w1d 17:42:45      Metric2: 1
                Validation State: unverified
                Task: BGP_10.2.1.1.4+55190
                AS path: I
                Communities: target:65002:1
                Import Accepted DeletePending
                VPN Label: 299888
                Localpref: 100
                Router ID: 10.1.1.4
                Secondary Tables: red.inet.0
```

show route label detail (Multipoint LDP Inband Signaling for Point-to-Multipoint LSPs)

```
user@host> show route label 299872 detail
mpls.0: 13 destinations, 13 routes (13 active, 0 holddown, 0 hidden)
299872 (1 entry, 1 announced)
        *LDP    Preference: 9
                Next hop type: Flood
                Next-hop reference count: 3
                Address: 0x9097d90
                Next hop: via vt-0/1/0.1
                Next-hop index: 661
                Label operation: Pop
                Address: 0x9172130
                Next hop: via so-0/0/3.0
                Next-hop index: 654
                Label operation: Swap 299872
                State: **Active Int>
                Local AS:  1001
                Age: 8:20       Metric: 1
                Task: LDP
                Announcement bits (1): 0-KRT
                AS path: I
                FECs bound to route: P2MP root-addr 10.255.72.166, grp 232.1.1.1, src
192.168.142.2
```

show route label detail (Multipoint LDP with Multicast-Only Fast Reroute)

```
user@host> show route label 301568 detail

mpls.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
301568 (1 entry, 1 announced)
        *LDP    Preference: 9
                Next hop type: Flood
                Address: 0x2735208
                Next-hop reference count: 3
                Next hop type: Router, Next hop index: 1397
                Address: 0x2735d2c
                Next-hop reference count: 3
                Next hop: 1.3.8.2 via ge-1/2/22.0
                Label operation: Pop
                Load balance label:  None;
                Next hop type: Router, Next hop index: 1395
                Address: 0x2736290
                Next-hop reference count: 3
                Next hop: 1.3.4.2 via ge-1/2/18.0
                Label operation: Pop
                Load balance label:  None;
                State: <Active Int AckRequest MulticastRPF>
                Local AS:    10
                Age: 54:05      Metric: 1
                Validation State: unverified
                Task: LDP
                Announcement bits (1): 0-KRT
                AS path: I
                FECs bound to route: P2MP root-addr 172.16.1.1, grp: 232.1.1.1, src:
192.168.219.11
                Primary Upstream : 172.16.1.3:0--172.16.1.2:0
                   RPF Nexthops  :
                        ge-1/2/15.0, 10.1.94.1, Label: 301568, weight: 0x1
                        ge-1/2/14.0, 10.1.3.1, Label: 301568, weight: 0x1
                Backup Upstream : 172.16.1.3:0--172.16.1.6:0
                   RPF Nexthops :
                        ge-1/2/20.0, 10.1.96.1, Label: 301584, weight: 0xfffe
                        ge-1/2/19.0, 10.1.6.1, Label: 301584, weight: 0xfffe
```

show route detail (Flexible VXLAN Tunnel Profile)

```
user@host> show route 192.168.0.2 detail
...

CUSTOMER_0001.inet.0: 5618 destinations, 6018 routes (5618 active, 0 holddown, 0 hidden)

192.168.0.2/32 (1 entry, 1 announced)
        *Static Preference: 5/100
                Next hop type: Router, Next hop index: 74781
                Address: 0x5d9b03cc
                Next-hop reference count: 363
                Next hop: via fti0.6, selected
                Session Id: 0x24c8
                State: <Active Int NSR-incapable OpaqueData Programmed>
                Age: 1:25:53
                Validation State: unverified
                        Tag: 10000001   Tag2: 1
                Announcement bits (2): 1-KRT 3-Resolve tree 30
                AS path: I
                Flexible IPv6 VXLAN tunnel profile
                    Action: Encapsulate
                    Interface: fti0.6 (Index: 10921)
                    VNI: 10000001
                    Source Prefix: 2001:db8:255::2/128
                    Source UDP Port Range: 54614 - 60074
                    Source MAC Address: 00:00:5e:00:52:01
                    Destination Address: 2001:db8:10:1:1:1:0:1
                    Destination UDP Port: 4790
                    VXLAN Flags: 0x08
...
```

Release Information

Command introduced before Junos OS Release 7.4.

Comand introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

`DeletePending` flag added to the command output in Junos OS Release 19.4R1.

`ipv6-origin`, `ipv6-target`, and `ipv6-extended Community` output fields supported in Junos OS Release and Junos OS Evolved Release 23.1.

show route exact

IN THIS SECTION

Syntax

```
show route exact destination-prefix
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route exact destination-prefix
<brief | detail | extensive | terse>
```

Description

Display only the routes that exactly match the specified address or range of addresses.

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`.
destination-prefix	Address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 exact

```
user@host> show route exact 207.17.136.0/24

inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both
```

```
207.17.136.0/24    *[Static/5] 2d 03:30:22
                    > to 192.168.71.254 via fxp0.0
```

show route exact detail

```
user@host> show route exact 207.17.136.0/24 detail

inet.0: 24 destinations, 25 routes (23 active, 0 holddown, 1 hidden)
Restart Complete
207.17.136.0/24 (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: 2d 3:30:26
                Task: RT
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I
```

show route exact extensive

```
user@host> show route exact 207.17.136.0/24 extensive
inet.0: 22 destinations, 23 routes (21 active, 0 holddown, 1 hidden)
207.17.136.0/24 (1 entry, 1 announced)
TSI:
KRT in-kernel 207.17.136.0/24 -> {192.168.71.254}
        *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: 1:25:18
                Task: RT
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I
```

show route exact terse

```
user@host> show route exact 207.17.136.0/24 terse

inet.0: 22 destinations, 23 routes (21 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 207.17.136.0/24    S   5                       >192.168.71.254
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route export

IN THIS SECTION

Syntax

```
show route export
<brief | detail>
<instance <instance-name> | routing-table-name>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route export
<brief | detail>
<instance <instance-name> | routing-table-name>
```

Description

Display policy-based route export information. Policy-based export simplifies the process of exchanging route information between routing instances.

Options

none	(Same as `brief`.) Display standard information about policy-based export for all instances and routing tables on all systems.
brief | detail	(Optional) Display the specified level of output.

instance <*instance-name*> (Optional) Display a particular routing instance for which policy-based export is currently enabled.

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

routing-table-name (Optional) Display information about policy-based export 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 export inet` command).

Required Privilege Level

view

Output Fields

Table 40 on page 685 lists the output fields for the `show route export` command. Output fields are listed in the approximate order in which they appear.

Table 40: show route export Output Fields

Field Name	Field Description	Level of Output
Table or *table-name*	Name of the routing tables that either import or export routes.	All levels
Routes	Number of routes exported from this table into other tables. If a particular route is exported to different tables, the counter will only increment by one.	`brief` none
Export	Whether the table is currently exporting routes to other tables: Y or N (Yes or No).	`brief` none
Import	Tables currently importing routes from the originator table. (Not displayed for tables that are not exporting any routes.)	`detail`

Table 40: show route export Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Flags	(instance keyword only) Flags for this feature on this instance: • config auto-policy—The policy was deduced from the configured IGP export policies. • cleanup—Configuration information for this instance is no longer valid. • config—The instance was explicitly configured.	detail
Options	(instance keyword only) Configured option displays the type of routing tables the feature handles: • unicast—Indicates *instance*.inet.0. • multicast—Indicates *instance*.inet.2. • unicast multicast—Indicates *instance*.inet.0 and *instance*.inet.2.	detail
Import policy	(instance keyword only) Policy that route export uses to construct the import-export matrix. Not displayed if the instance type is vrf.	detail
Instance	(instance keyword only) Name of the routing instance.	detail
Type	(instance keyword only) Type of routing instance: forwarding, non-forwarding, or vrf.	detail

Sample Output

show route export

```
user@host> show route export
Table                            Export          Routes
inet.0                           N                    0
```

```
black.inet.0                      Y                     3
red.inet.0                        Y                     4
```

show route export detail

```
user@host> show route export detail
inet.0                                          Routes:        0
black.inet.0                                    Routes:        3
  Import: [ inet.0 ]
red.inet.0                                      Routes:        4
  Import: [ inet.0 ]
```

show route export instance detail

```
user@host> show route export instance detail
Instance: master                        Type: forwarding
  Flags: <config auto-policy> Options: <unicast multicast>
  Import policy: [ (ospf-master-from-red || isis-master-from-black) ]
Instance: black                         Type: non-forwarding
Instance: red                           Type: non-forwarding
```

Release Information

Command introduced before Junos OS Release 7.4.

show route extensive

IN THIS SECTION

Syntax

```
show route extensive
<destination-prefix>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route extensive
<destination-prefix>
```

Description

Display extensive information about the active entries in the routing tables.

Options

none	Display all active entries in the routing table.

destination-prefix	(Optional) Display active entries for the specified address or range of addresses.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 41 on page 689 describes the output fields for the `show route extensive` command. Output fields are listed in the approximate order in which they appear.

Table 41: show route extensive Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).
number destinations	Number of destinations for which there are routes in the routing table.
number routes	Number of routes in the routing table and total number of routes in the following states: • `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 41: show route extensive Output Fields *(Continued)*

Field Name	Field Description

route-destination (entry, announced)	Route destination (for example: 10.0.0.1/24). The `entry` value is the number of route 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:
	• *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).
	  ◦ *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`.

TSI	Protocol header information.
label stacking	(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. • `S=0 route` indicates that a packet with an incoming label stack depth of two or more exits this router 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).

Table 41: show route extensive Output Fields *(Continued)*

Field Name	Field Description
[*protocol*, *preference*]	Protocol from which the route was learned and the preference value for the route. • `+`—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. 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.
Level	(IS-IS only). In IS-IS, a single autonomous system (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.
Route Distinguisher	IP subnet augmented with a 64-bit prefix.
PMSI	Provider multicast service interface (MVPN routing table).
Next-hop type	Type of next hop.
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.

Table 41: show route extensive Output Fields *(Continued)*

Field Name	Field Description
Source	IP address of the route source.
Next hop	Network layer address of the directly reachable neighboring system.
via	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: • 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).
Offset	Whether the metric has been increased or decreased by an offset value.
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 recursively derive a forwarding 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).

Table 41: show route extensive Output Fields *(Continued)*

Field Name	Field Description
Indirect next hops	When present, a list of nodes that are used to resolve the path to the next-hop destination, in the order that they are resolved. When BGP PIC Edge is enabled, the output lines that contain `Indirect next hop: weight` follow next hops that the software can use to repair paths where a link failure occurs. The next-hop weight has one of the following values: • 0x1 indicates active next hops. • 0x4000 indicates passive next hops.
State	State of the route (a route can be in more than one state).
Session ID	The BFD session ID number that represents the protection using MPLS fast reroute (FRR) and loop-free alternate (LFA).
Weight	Weight for the backup path. If the weight of an indirect next hop is larger than zero, the weight value is shown.

Table 41: show route extensive Output Fields *(Continued)*

Field Name	Field Description
Inactive reason	If the route is inactive, the reason for its current state is indicated. Typical reasons include: • `Active preferred`—Currently active route was selected over this route. • `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. • `Cluster list length`—Path with a shorter cluster list length is available. • `Forwarding use only`—Path is only available for forwarding purposes. • `IGP metric`—Path through the next hop with a lower IGP metric is available. • `IGP metric type`—Path with a lower OSPF link-state advertisement type is available. • `Interior > Exterior > Exterior via Interior`—Direct, static, IGP, or EBGP path is available. • `Local preference`—Path with a higher local preference value is available. • `Next hop address`—Path with a lower metric next hop is available. • `No difference`—Paths from all neighbors are identical for all necessary route attributes in some special scenarios, such as BGP L2VPN/VPLS, and there is no difference. • `Not Best in its group`—Occurs when multiple peers of the same external AS advertise the same prefix and are grouped together in the selection process. When this reason is displayed, an additional reason is provided (typically one of the other reasons listed). • `Number of gateways`—Path with a higher number of next hops is available. • `Origin`—Path with a lower origin code is available. • `OSPF version`—Path does not support the indicated OSPF version. • `RIB preference`—Route from a higher-numbered routing table is available. • `Route destinguisher`—64-bit prefix added to IP subnets to make them unique.

Table 41: show route extensive Output Fields *(Continued)*

Field Name	Field Description
	• `Route metric or MED comparison`—Route with a lower metric or MED is available. • `Route preference`—Route with a lower preference value is available. • `Router ID`—Path through a neighbor with a lower ID is available. • `Unusable path`—Path is not usable because of one of the following conditions: the route is damped, the route is rejected by an import policy, or the route is unresolved. • `Update source`—Last tiebreaker is the lowest IP address value.
Local AS	Autonomous system (AS) number of the local routing device.
Age	How long the route has been known.
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.
Metric	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.
MED-plus-IGP	Metric value for BGP path selection to which the IGP cost to the next-hop destination has been added.
TTL-Action	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.
Task	Name of the protocol that has added the route.

Table 41: show route extensive Output Fields *(Continued)*

Field Name	Field Description
Announcement bits	List of protocols that are consumers of the route. Using the following output as an example, `Announcement bits (3): 0-KRT 5-Resolve tree 2 8-BGP RT Background` there are (3) announcement bits to reflect the three clients (protocols) that have state for this route: Kernel (0-KRT), 5 (resolution tree process 2), and 8 (BGP). The notation *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	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: • `I`—IGP. • `E`—EGP. • `Recorded`—The AS path is recorded by the sample process (sampled). • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—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. **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.

Table 41: show route extensive Output Fields *(Continued)*

Field Name	Field Description
validation-state	(BGP-learned routes) Validation status of the route: • `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. • `Unverified`—Indicates that origin validation is not enabled for the BGP peers. • `Valid`—Indicates that the prefix and autonomous system pair are found in the database.
FECs bound to route	Point-to-multipoint root address, multicast source address, and multicast group address when multipoint LDP (M-LDP) inband signaling is configured.
AS path: I <Originator>	(For route reflected output only) Originator ID attribute set by the route reflector.
route status	Indicates the status of a BGP route: • `Accepted`—The specified BGP route is imported by the default BGP policy. • `Import`—The route is imported into a Layer 3 VPN routing instance. • `Import-Protect`—A remote instance egress that is protected. • `Multipath`—A BGP multipath active route. • `MultipathContrib`—The route is not active but contributes to the BGP multipath. • `Protect`—An egress route that is protected. • `Stale`—A route that is marked stale due to graceful restart.
Primary Upstream	When multipoint LDP with multicast-only fast reroute (MoFRR) is configured, the primary upstream path. MoFRR transmits a multicast join message from a receiver toward a source on a primary path, while also transmitting a secondary multicast join message from the receiver toward the source on a backup path.

Table 41: show route extensive Output Fields *(Continued)*

Field Name	Field Description
RPF Nexthops	When multipoint LDP with MoFRR is configured, the reverse-path forwarding (RPF) next-hop information. Data packets are received from both the primary path and the secondary paths. The redundant packets are discarded at topology merge points due to the RPF checks.
Label	Multiple MPLS labels are used to control MoFRR stream selection. Each label represents a separate route, but each references the same interface list check. Only the primary label is forwarded while all others are dropped. Multiple interfaces can receive packets using the same label.
weight	Value used to distinguish MoFRR primary and backup routes. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible.
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.
Cluster list	(For route reflected output only) Cluster ID sent by the route reflector.
Originator ID	(For route reflected output only) Address of router that originally sent the route to the route reflector.
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.
DeletePending	The DeletePending flag indicates that a BGP route needs to be processed due to a BGP peer down event.
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).

Table 41: show route extensive 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).
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.
Originating RIB	Name of the routing table whose active route was used to determine the forwarding next-hop entry in the resolution database. For example, in the case of inet.0 resolving through inet.0 and inet.3, this field indicates which routing table, inet.0 or inet.3, provided the best path for a particular prefix.
Node path count	Number of nodes in the path.
Forwarding nexthops	Number of forwarding next hops. The forwarding next hop is the network layer address of the directly reachable neighboring system (if applicable) and the interface used to reach it.
Statistics ID Group	Indicates the Kernel ID number and Statistics ID number.
Statistics	Indicates the number of packets and data transferred.

Sample Output

show route extensive

```
user@host> show route extensive
inet.0: 22 destinations, 23 routes (21 active, 0 holddown, 1 hidden)
203.0.113.10/16 (1 entry, 1 announced)
TSI:
KRT in-kernel 203.0.113.10/16 -> {192.168.71.254}
        *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:    64496
                Age: 1:34:06
                Task: RT
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

203.0.113.30/30 (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:    64496
                Age: 1:32:40
                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:    64496
                Age: 1:32:40    Metric: 1
                Area: 0.0.0.0
                Task: OSPF
                AS path: I
```

```
203.0.113.103/32 (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:    644969
                Age: 1:32:43
                Task: IF
                Announcement bits (1): 3-Resolve tree 2
                AS path: I

...

203.0.113.203/30 (1 entry, 1 announced)
TSI:
KRT in-kernel 203.0.113.203/30 -> {203.0.113.216}
        *OSPF   Preference: 10
                Next-hop reference count: 9
                Next hop: via so-0/3/0.0
                Next hop: 203.0.113.216 via ge-3/1/0.0, selected
                State: <Active Int>
                Local AS:    64496
                Age: 1:32:19    Metric: 2
                Area: 0.0.0.0
                Task: OSPF
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I

...

198.51.100.2/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 198.51.100.2/32 -> {}
        *PIM    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:08
                Task: PIM Recv
                Announcement bits (2): 0-KRT 3-Resolve tree 2
                AS path: I
```

```
...

198.51.100.22/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 198.51.100.22/32 -> {}
        *IGMP   Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:06
                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)

203.0.113.103/32 (1 entry, 1 announced)
        State: <FlashAll>
        *RSVP   Preference: 7
                Next-hop reference count: 6
                Next hop: 203.0.113.216 via ge-3/1/0.0 weight 0x1, selected
                Label-switched-path green-r1-r3
                Label operation: Push 100096
                State: <Active Int>
                Local AS:    64496
                Age: 1:28:12    Metric: 2
                Task: RSVP
                Announcement bits (2): 1-Resolve tree 1 2-Resolve tree 2
                AS path: I

203.0.113.238/32 (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:    64496
                Age: 1:28:12    Metric: 1
                Task: RSVP
                Announcement bits (2): 1-Resolve tree 1 2-Resolve tree 2
                AS path: I
```

```
private1__.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:    64496
                Age: 1:34:07
                Task: IF
                AS path: I

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

0   (1 entry, 1 announced)
TSI:
KRT in-kernel 0      /36 -> {}
        *MPLS   Preference: 0
                Next hop type: Receive
                Next-hop reference count: 6
                State: <Active Int>
                Local AS:    64496
                Age: 1:34:08    Metric: 1
                Task: MPLS
                Announcement bits (1): 0-KRT
                AS path: I

...

mpls.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
  299840 (1 entry, 1 announced)
  TSI:
  KRT in-kernel 299840 /52 -> {indirect(1048575)}
          *RSVP   Preference: 7/2
                  Next hop type: Flood
                  Address: 0x9174a30
                  Next-hop reference count: 4
                  Next hop type: Router, Next hop index: 798
```

```
                Address: 0x9174c28
                Next-hop reference count: 2
                Next hop: 198.51.100.2 via lt-1/2/0.9 weight 0x1
                Label-switched-path R2-to-R4-2p2mp
                Label operation: Pop
                Next hop type: Router, Next hop index: 1048574
                Address: 0x92544f0
                Next-hop reference count: 2
                Next hop: 198.51.100.2 via lt-1/2/0.7 weight 0x1
                Label-switched-path R2-to-R200-p2mp
                Label operation: Pop
                Next hop: 198.51.100.2 via lt-1/2/0.5 weight 0x8001
                Label operation: Pop
                State: <Active Int>
                Age: 1:29       Metric: 1
                Task: RSVP
                Announcement bits (1): 0-KRT
                AS path: I...

800010 (1 entry, 1 announced)

TSI:
KRT in-kernel 800010 /36 -> {vt-3/2/0.32769}
        *VPLS   Preference: 7
                Next-hop reference count: 2
                Next hop: via vt-3/2/0.32769, selected
                Label operation: Pop
                State: <Active Int>
                Age: 1:31:53
                Task: Common L2 VC
                Announcement bits (1): 0-KRT
                AS path: I

vt-3/2/0.32769 (1 entry, 1 announced)
TSI:
KRT in-kernel vt-3/2/0.32769.0      /16 -> {indirect(1048574)}
        *VPLS   Preference: 7
                Next-hop reference count: 2
                Next hop: 203.0.113.216 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: 203.0.113.103
                Push 800012
```

```
                Indirect next hop: 87272e4 1048574
                State: <Active Int>
                Age: 1:31:53    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
                Indirect next hops: 1
                        Protocol next hop: 203.0.113.103 Metric: 2
                        Push 800012
                        Indirect next hop: 87272e4 1048574
                        Indirect path forwarding next hops: 1
                                Next hop: 203.0.113.216 via ge-3/1/0.0 weight 0x1
                        203.0.113.103/32 Originating RIB: inet.3
                          Metric: 2                       Node path count: 1
                          Forwarding nexthops: 1
                                Nexthop: 203.0.113.216 via ge-3/1/0.0

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

2001:db8::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:    64496
                Age: 1:34:07
                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:    64496
                Age: 1:34:07
                Task: IF
                AS path: I
```

```
ff02::2/128 (1 entry, 1 announced)
TSI:
KRT in-kernel ff02::2/128 -> {}
        *PIM    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:08
                Task: PIM Recv6
                Announcement bits (1): 0-KRT
                AS path: I

ff02::d/128 (1 entry, 1 announced)
TSI:
KRT in-kernel ff02::d/128 -> {}
        *PIM    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:08
                Task: PIM Recv6
                Announcement bits (1): 0-KRT
                AS path: I

ff02::16/128 (1 entry, 1 announced)
TSI:
KRT in-kernel ff02::16/128 -> {}
        *MLD    Preference: 0
                Next-hop reference count: 18
                State: <Active NoReadvrt Int>
                Local AS:    64496
                Age: 1:34:06
                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: 1:34:07
                Task: IF
                AS path: I

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

203.0.113.103:1:3:1/96 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 203.0.113.103:1
                Next-hop reference count: 7
                Source: 203.0.113.103
                Protocol next hop: 203.0.113.103
                Indirect next hop: 2 no-forward
                State: <Secondary Active Int Ext>
                Local AS:    64496 Peer AS:    64496
                Age: 1:28:12    Metric2: 1
                Task: BGP_69.203.0.113.103+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: 203.0.113.103
                Primary Routing Table bgp.l2vpn.0

203.0.113.152:1:1:1/96 (1 entry, 1 announced)
TSI:
Page 0 idx 0 Type 1 val 8699540
        *L2VPN  Preference: 170/-1
                Next-hop reference count: 5
                Protocol next hop: 203.0.113.152
                Indirect next hop: 0 -
                State: <Active Int Ext>
                Age: 1:34:03    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
```

```
203.0.113.152:1:5:1/96 (1 entry, 1 announced)
TSI:
Page 0 idx 0 Type 1 val 8699528
        *L2VPN  Preference: 170/-101
                Next-hop reference count: 5
                Protocol next hop: 203.0.113.152
                Indirect next hop: 0 -
                State: <Active Int Ext>
                Age: 1:34:03    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

...

l2circuit.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

TSI:

203.0.113.163: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: 203.0.113.163 Indirect next hop: 86af000 296
                State: <Active Int>
                Local AS:    64499
                Age: 10:21
                Task: l2 circuit
                Announcement bits (1): 0-LDP
                AS path: I
                VC Label 100000, MTU 1500, VLAN ID 512


203.0.113.55/24 (1 entry, 1 announced)
TSI:
KRT queued (pending) add
  198.51.100.0/24 -> {Push 300112}
        *BGP    Preference: 170/-101
                Next hop type: Router
                Address: 0x925c208
```

```
                Next-hop reference count: 2
                Source: 203.0.113.9
                Next hop: 203.0.113.9 via ge-1/2/0.15, selected
                Label operation: Push 300112
                Label TTL action: prop-ttl
                State: <Active Ext>
                Local AS:  64509 Peer AS: 65539
                Age: 1w0d 23:06:56
                AIGP: 25
                Task: BGP_65539.203.0.113.9+56732
                Announcement bits (1): 0-KRT
                AS path: 65539 64508 I
                Accepted
                Route Label: 300112
                Localpref: 100
                Router ID: 213.0.113.99
```

show route programmed extensive (with statistics information)

```
user@host> show route programmed extensive

inet.0: 104 destinations, 104 routes (103 active, 0 holddown, 1 hidden)

inet.3: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
192.0.2.0/24 (1 entry, 1 announced)
        *Static Preference: 5/100
                Next hop type: Router, Next hop index: 0
                Address: 0xa667694
                Next-hop reference count: 15
                Next hop: 198.51.100.0 via ae0.0
                Label operation: Push 50008, Push 3400(top)
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3400: None;
                Label element ptr: 0x84e7e90
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 10, Stats IDs = { 536870923 }
                Statistics: Packets 1380, Bytes 681720
```

```
Next hop: 198.51.100.1 via ae0.1
Label operation: Push 50008, Push 3401(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3401: None;
Label element ptr: 0x84e7ee0
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 11, Stats IDs = { 536870924 }
Statistics: Packets 1444, Bytes 713336
Next hop: 198.51.100.2 via ae0.2
Label operation: Push 50008, Push 3410(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3410: None;
Label element ptr: 0xa1926a0
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 12, Stats IDs = { 536870925 }
Statistics: Packets 1420, Bytes 701480
Next hop: 198.51.100.3 via ge-0/0/2.0
Label operation: Push 50008, Push 3411(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3411: None;
Label element ptr: 0xa1926c8
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 13, Stats IDs = { 536870926 }
Statistics: Packets 1486, Bytes 734084
Next hop: 198.51.100.4 via ge-0/0/2.1, selected
Label operation: Push 50008, Push 3420(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3420: None;
Label element ptr: 0xa1926f0
Label parent element ptr: 0x0
Label element references: 3
```

```
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 14, Stats IDs = { 536870927 }
Statistics: Packets 1399, Bytes 691106
Next hop: 198.51.100.5 via ge-0/0/2.2
Label operation: Push 50008, Push 3421(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3421: None;
Label element ptr: 0xa192718
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 15, Stats IDs = { 536870928 }
Statistics: Packets 1455, Bytes 718770
Next hop: 198.51.100.6 via ge-0/0/4.0
Label operation: Push 50008, Push 3450(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3450: None;
Label element ptr: 0xa192740
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 16, Stats IDs = { 536870929 }
Statistics: Packets 1407, Bytes 695058
Next hop: 198.51.100.7 via ge-0/0/4.1
Label operation: Push 50008, Push 3460(top)
Label TTL action: prop-ttl, prop-ttl(top)
Load balance label: Label 50008: None; Label 3460: None;
Label element ptr: 0xa192768
Label parent element ptr: 0x0
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Statistics ID Group: Kernel ID = 17, Stats IDs = { 536870930 }
Statistics: Packets 1418, Bytes 700492
Next hop: 198.51.100.8 via ge-0/0/4.2
Label operation: Push 50008, Push 3470(top)
```

```
                Label TTL action: prop-ttl, prop-ttl(top)
                Load balance label: Label 50008: None; Label 3470: None;
                Label element ptr: 0xa192790
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Statistics ID Group: Kernel ID = 18, Stats IDs = { 536870931 }
                Statistics: Packets 0, Bytes 0
                State: <Active Int NSR-incapable Programmed>
                Age: 12:46
                Validation State: unverified
                Announcement bits (3): 0-Resolve tree 6 1-Resolve tree 1 3-Resolve_IGP_FRR task
                AS path: I
                Session-IDs associated:
                Session-id: 370 Version: 0
                Thread: junos-main
```

show route extensive (BGP-SRTE routes)

```
user@host> show route extensive
inet.0: 22 destinations, 23 routes (21 active, 0 holddown, 1 hidden)
9.9.9.9-1 <c>/64 (1 entry, 0 announced):
        **SPRING-TE Preference: 8
                Next hop type: Indirect, Next hop index: 0
                Address: 0xdc33080
                Next-hop reference count: 1
                Next hop type: Router, Next hop index: 0
                Next hop: 1.2.2.2 via ge-0/0/2.0, selected
                Label element ptr: 0xdf671d0
                Label parent element ptr: 0x0
                Label element references: 11
                Label element child references: 0
                Label element lsp id: 0
                Session Id: 0x0
                Protocol next hop: 299920
                Label operation: Push 800040
                Label TTL action: prop-ttl
                Load balance label: Label 800040: None;
                Composite next hop: 0xcd4f950 - INH Session ID: 0x0
```

```
                  Indirect next hop: 0xdc99a84 - INH Session ID: 0x0 Weight 0x1
                  State: <Active Int>
                  Local AS:   100
                  Age: 5d 17:37:19     Metric: 1     Metric2: 16777215
                  Validation State: unverified
                  Task: SPRING-TE
                  AS path:
                  SRTE Policy State:
                  SR Preference/Override: 200/100
                  Tunnel Source: Static configuration
                  Composite next hops: 1
                          Protocol next hop: 299920 Metric: 0
                          Label operation: Push 800040
                          Label TTL action: prop-ttl
                          Load balance label: Label 800040: None;
                          Composite next hop: 0xcd4f950 - INH Session ID: 0x0
                          Indirect next hop: 0xdc99a84 - INH Session ID: 0x0 Weight 0x1
                          Indirect path forwarding next hops: 1
                                  Next hop type: Router
                                  Next hop: 1.2.2.2 via ge-0/0/2.0
                                  Session Id: 0x0
                                  299920 /52 Originating RIB: mpls.0
                                    Metric: 0 Node path count: 1
                                    Forwarding nexthops: 1
                                          Next hop type: Router
                                          Next hop: 1.2.2.2 via ge-0/0/2.0
                                          Session Id: 0x141
```

Release Information

Command introduced before Junos OS Release 7.4.

`DeletePending` flag added to the command output in Junos OS Release 19.4R1.

show route forwarding-table

IN THIS SECTION

Syntax

```
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>
```

Syntax (MX Series Routers)

```
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>
```

Syntax (TX Matrix and TX Matrix Plus Routers)

```
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>
```

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: `bridge (ccc | destination | detail | extensive | interface-name | label | learning-vlan-id | matching | multicast | summary | table | vlan | vpn)`, `ethernet-switching`, `evpn`, `fibre-channel`, `fmembers`, `inet`, `inet6`, `iso`, `mcsnoop-inet`, `mcsnoop-inet6`, `mpls`, `satellite-inet`, `satellite-inet6`, `satellite-vpls`, `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*	(TX Matrix and TX matrix Plus routers only) (Optional) On a routing matrix composed of a TX Matrix router and T640 routers, 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 or T4000 routers, display information for the specified router (line-card chassis) connected to the TX Matrix Plus router. Replace *number* with the following values depending on the LCC configuration: • 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix. • 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix. • 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix. • 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
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	(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

Output Fields

Table 42 on page 718 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 42: 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` *`logical-system-name/routing-instance-name`* 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

Table 42: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Enabled protocols	The features and protocols that have been enabled for a given routing table. This field can contain the following values: • BUM hashing—BUM hashing is enabled. • MAC Stats—Mac Statistics is enabled. • Bridging—Routing instance is a normal layer 2 bridge. • No VLAN—No VLANs are associated with the bridge domain. • All VLANs—The `vlan-id all` statement has been enabled for this bridge domain. • Single VLAN—Single VLAN ID is associated with the bridge domain. • MAC action drop—New MACs will be dropped when the MAC address limit is reached. • Dual VLAN—Dual VLAN tags are associated with the bridge domain • No local switching—No local switching is enabled for this routing instance.. • Learning disabled—Layer 2 learning is disabled for this routing instance. • MAC limit reached—The maximum number of MAC addresses that was configured for this routing instance has been reached. • VPLS—The VPLS protocol is enabled. • No IRB l2-copy—The no-irb-layer-2-copy feature is enabled for this routing instance. • ACKed by all peers—All peers have acknowledged this routing instance. • BUM Pruning—BUM pruning is enabled on the VPLS instance. • Def BD VXLAN—VXLAN is enabled for the default bridge domain. • EVPN—EVPN protocol is enabled for this routing instance.	All levels

Table 42: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
	• Def BD OVSDB—Open vSwitch Database (OVSDB) is enabled on the default bridge domain. • Def BD Ingress replication—VXLAN ingress node replication is enabled on the default bridge domain. • L2 backhaul—Layer 2 backhaul is enabled. • FRR optimize—Fast reroute optimization • MAC pinning—MAC pinning is enabled for this bridge domain. • MAC Aging Timer—The MAC table aging time is set per routing instance. • EVPN VXLAN—This routing instance supports EVPN with VXLAN encapsulation. • PBBN—This routing instance is configured as a provider backbone bridged network. • PBN—This routing instance is configured as a provider bridge network. • ETREE—The ETREE protocol is enabled on this EVPN routing instance. • ARP/NDP suppression—EVPN ARP NDP suppression is enabled in this routing instance. • Def BD EVPN VXLAN—EVPN VXLAN is enabled for the default bridge domain. • MPLS control word—Control word is enabled for this MPLS routing instance.	
Address family	Address family (for example, `IP`, `IPv6`, `ISO`, `MPLS`, and `VPLS`).	All levels
Destination	Destination of the route.	`detail` `extensive`

Table 42: 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): • `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`

Table 42: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Flags	Route type flags: • `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. **NOTE**: For static routes that use point-to-point (P2P) outgoing interfaces, the next-hop address is not displayed in the output.	`detail extensive`

Table 42: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Next hop Type (Type)	Next-hop type. When the `detail` keyword is used, the next-hop type might be abbreviated (as indicated in parentheses): • `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 (mcrt)`—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 (recv)`—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. • `VxLAN Local`—EVPN Type 5 route in kernel.	`detail extensive`
Index	Software index of the next hop that is used to route the traffic for a given prefix.	`detail extensive` none

Table 42: show route forwarding-table Output Fields *(Continued)*

Field Name	Field Description	Level of Output
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`
sftw	On Junos OS Evolved devices, `sftw` is another type of next-hop. A software next-hop can contain a link to another single next-hop, and hence it often works as an indirect next-hop. The usage of `sftw` helps to reduce churn in Junos OS Evolved objects by providing a common object where topology updates can be reflected. Further MPLS states can also be attached to these software next-hops. When a route is created, a `sftw` next-hop seems to be created whether it is the management port or the loopback interface.	`detail` `extensive` none

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
172.16.1.0/24         ifdn     0                    rslv   608     1 ge-2/0/1.0
172.16.1.0/32         iddn     0 172.16.1.0            recv   606     1 ge-2/0/1.0
172.16.1.1/32         user     0                    rjct    46     4
172.16.1.1/32         intf     0 172.16.1.1            locl   607     2
172.16.1.1/32         iddn     0 172.16.1.1            locl   607     2
172.16.1.255/32       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           locl   615     2
10.0.0.1/32        dest     0 10.0.0.1           locl   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           locl   611     2
10.1.1.1/32        iddn     0 10.1.1.1           locl   611     2
10.1.1.255/32      iddn     0 ff:ff:ff:ff:ff:ff  bcst   609     1 ge-2/0/1.0
10.206.0.0/16      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       locl   417     2
10.209.2.131/32    dest     0 10.209.2.131       locl   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                   locl    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           locl   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         locl   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          locl   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           locl   135     2
10.0.0.4/32        dest     0 10.0.0.4           locl   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 (EVPN Type 5 route with Type 2 and Type 5 route coexistence)

```
user@device> show route forwarding-table destination 10.1.1.20 table vrf1 extensive
Routing table: vrf1.inet [Index 9]
Internet:
Destination:  10.1.1.20/32
  Route type: user
  Route reference: 0                   Route interface-index: 0
  Multicast RPF nh index: 0
  P2mpidx: 0
  Flags: sent to PFE, VxLAN Local
  Nexthop:
```

```
  Next-hop type: composite            Index: 2694     Reference: 7
  Next-hop type: indirect             Index: 524326   Reference: 2
  Next-hop type: unilist              Index: 524288   Reference: 5
  Nexthop: 10.1.1.1
  Next-hop type: unicast              Index: 1724     Reference: 15
  Next-hop interface: xe-0/0/1.0    Weight: 0x0
  Nexthop: 10.1.1.4  Next-hop type: unicast              Index: 1725     Reference: 15
  Next-hop interface: xe-0/0/4.0    Weight: 0x0
```

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 192.0.2.2/30;
        }
    }
}
```

show route forwarding-table (Junos OS Evolved devices with next-hop type 'sftw')

```
user@host> show route forwarding-table | grep sftw
Internet:
Destination        Type RtRef Next hop           Type Index NhRef Netif
default            user     0                    sftw     7021     1 re0:mgmt-0.0
10.21.1.0/32       user     0                    sftw     7022     1 et-0/0/32.0

user@host> show route forwarding-table | grep sftw
Internet:
Destination        Type RtRef Next hop           Type Index NhRef Netif
default            user     0                    sftw     5021     1 re0:mgmt-0.0
10.22.1.0/32       user     0                    sftw     5022     1 et-0/0/32.0
```

Release Information

Command introduced before Junos OS Release 7.4.

Option `bridge-domain` introduced in Junos OS Release 7.5

Option `learning-vlan-id` introduced in Junos OS Release 8.4

Options `all` and `vlan` introduced in Junos OS Release 9.6.

RELATED DOCUMENTATION

show route instance

show route hidden

IN THIS SECTION

Syntax

```
show route hidden
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

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

Required Privilege Level

view

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.l3vpn.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 731.

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                       >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

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.l3vpn.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)
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

Understanding Hidden Routes

show route inactive-path

IN THIS SECTION

Syntax

```
show route inactive-path
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route inactive-path
<brief | detail | extensive | terse>
```

Description

Display routes for destinations that have no active route. An inactive route is a route that was not selected as the best path.

Options

none	Display all inactive routes.
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 | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 inactive-path

```
user@host> show route inactive-path

inet.0: 25 destinations, 26 routes (24 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both

10.12.100.12/30     [OSPF/10] 03:57:28, metric 1
                    > via so-0/3/0.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
10.0.0.0/8          [Direct/0] 04:39:56
                    > via fxp1.0

red.inet.0: 6 destinations, 8 routes (4 active, 0 holddown, 3 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both

10.12.80.0/30       [BGP/170] 04:38:17, localpref 100
                      AS path: 100 I
                    > to 10.12.80.1 via ge-6/3/2.0

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.l3vpn.0: 3 destinations, 3 routes (0 active, 0 holddown, 3 hidden)
Restart Complete

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 inactive-path detail

```
user@host> show route inactive-path detail

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

10.12.100.12/30 (2 entries, 1 announced)
         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:     1
                Age: 3:58:24    Metric: 1
```

```
                Area: 0.0.0.0
                Task: OSPF
                AS path: I

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

10.0.0.0/8 (2 entries, 0 announced)
         Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 1
                Next hop: via fxp1.0, selected
                State: <NotBest Int>
                Inactive reason: No difference
                Age: 4:40:52
                Task: IF
                AS path: I

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

10.12.80.0/30 (2 entries, 1 announced)
         BGP    Preference: 170/-101
                Next-hop reference count: 6
                Source: 10.12.80.1
                Next hop: 10.12.80.1 via ge-6/3/2.0, selected
                State: <Ext>
                Inactive reason: Route Preference
                Peer AS:   100
                Age: 4:39:13
                Task: BGP_100.10.12.80.1+179
                AS path: 100 I
                Localpref: 100
                Router ID: 10.0.0.0
```

show route inactive-path extensive

The output for the `show route inactive-path extensive` command is identical to that of the `show route inactive-path detail` command.

show route inactive-path terse

```
user@host> show route inactive-path 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
  10.12.100.12/30    O  10          1            >so-0/3/0.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
  10.0.0.0/8         D   0                       >fxp1.0

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
  10.12.80.0/30      B 170        100            >10.12.80.1      100 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.l3vpn.0: 3 destinations, 3 routes (0 active, 0 holddown, 3 hidden)
Restart Complete

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)
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route active-path

show route detail

show route extensive

show route terse

show route instance

IN THIS SECTION

Syntax

```
show route instance
<brief | detail | summary>
<instance-name>
```

```
<logical-system (all | logical-system-name)>
<operational>
```

Syntax (EX Series Switches and QFX Series)

```
show route instance
<brief | detail | summary>
<instance-name>
<operational>
```

Description

Display routing instance information.

Options

none	(Same as `brief`) Display standard information about all routing instances.
brief | detail | summary	(Optional) Display the specified level of output. If you do not specify a level of output, the system defaults to `brief`. (These options are not available with the `operational` keyword.)
instance-name	(Optional) Display information for all routing instances whose name begins with this string (for example, `cust1`, `cust11`, and `cust111` are all displayed when you run the `show route instance cust1` command).
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
operational	(Optional) Display operational routing instances.

Required Privilege Level

view

Output Fields

Table 43 on page 741 lists the output fields for the `show route instance` command. Output fields are listed in the approximate order in which they appear.

Table 43: show route instance Output Fields

Field Name	Field Description	Level of Output
Instance or *instance-name*	Name of the routing instance.	All levels
Operational Routing Instances	(`operational` keyword only) Names of all operational routing instances.	—
Type	Type of routing instance: `forwarding`, `l2vpn`, `no-forwarding`, `vpls`, `virtual-router`, or `vrf`.	All levels
State	State of the routing instance: `active` or `inactive`.	`brief` `detail` none
Interfaces	Name of interfaces belonging to this routing instance.	`brief` `detail` none
Restart State	Status of graceful restart for this instance: `Pending` or `Complete`.	`detail`
Path selection timeout	Maximum amount of time, in seconds, remaining until graceful restart is declared complete. The default is `300`.	`detail`
Tables	Tables (and number of routes) associated with this routing instance.	`brief` `detail` none
Route-distinguisher	Unique route distinguisher associated with this routing instance.	`detail`

Table 43: show route instance Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Vrf-import	VPN routing and forwarding instance import policy name.	`detail`
Vrf-export	VPN routing and forwarding instance export policy name.	`detail`
Vrf-import-target	VPN routing and forwarding instance import target community name.	`detail`
Vrf-export-target	VPN routing and forwarding instance export target community name.	`detail`
Vrf-edge-protection-id	Context identifier configured for edge-protection.	`detail`
Fast-reroute-priority	Fast reroute priority setting for a VPLS routing instance: `high`, `medium`, or `low`. The default is `low`.	`detail`
Restart State	Restart state: • `Pending:`*`protocol-name`*—List of protocols that have not yet completed graceful restart for this routing table. • `Complete`—All protocols have restarted for this routing table.	`detail`
Primary rib	Primary table for this routing instance.	`brief` none `summary`
Active/holddown/ hidden	Number of active, hold-down, and hidden routes.	All levels

Sample Output

show route instance

```
user@host> show route instance
Instance             Type
         Primary RIB                                     Active/holddown/hidden
master               forwarding
         inet.0                                          16/0/1
         iso.0                                           1/0/0
         mpls.0                                          0/0/0
         inet6.0                                         2/0/0
         l2circuit.0                                     0/0/0
__juniper_private1__ forwarding
         __juniper_private1__.inet.0                     12/0/0
         __juniper_private1__.inet6.0                    1/0/0
```

show route instance detail (VPLS Routing Instance)

```
user@host> show route instance detail test-vpls
test-vpls:
  Router ID: 0.0.0.0
  Type: vpls             State: Active
  Interfaces:
    lsi.1048833
    lsi.1048832
    fe-0/1/0.513
  Route-distinguisher: 10.255.37.65:1
  Vrf-import: [ __vrf-import-test-vpls-internal__ ]
  Vrf-export: [ __vrf-export-test-vpls-internal__ ]
  Vrf-import-target: [ target:300:1 ]
  Vrf-export-target: [ target:300:1 ]
  Vrf-edge-protection-id:  166.1.3.1  Fast-reroute-priority: high
  Tables:
    test-vpls.l2vpn.0          : 3 routes (3 active, 0 holddown, 0 hidden)
```

show route instance operational

```
user@host> show route instance operational
Operational Routing Instances:

master
default
```

show route instance summary

```
user@host> show route instance summary
Instance            Type           Primary rib            Active/holddown/hidden
master              forwarding
                                   inet.0                 15/0/1
                                   iso.0                  1/0/0
                                   mpls.0                 35/0/0
                                   l3vpn.0                0/0/0
                                   inet6.0                2/0/0
                                   l2vpn.0                0/0/0
                                   l2circuit.0            0/0/0
BGP-INET            vrf
                                   BGP-INET.inet.0        5/0/0
                                   BGP-INET.iso.0         0/0/0
                                   BGP-INET.inet6.0       0/0/0
BGP-L               vrf
                                   BGP-L.inet.0           5/0/0
                                   BGP-L.iso.0            0/0/0
                                   BGP-L.mpls.0           4/0/0
                                   BGP-L.inet6.0          0/0/0
L2VPN               l2vpn
                                   L2VPN.inet.0           0/0/0
                                   L2VPN.iso.0            0/0/0
                                   L2VPN.inet6.0          0/0/0
                                   L2VPN.l2vpn.0          2/0/0
LDP                 vrf
                                   LDP.inet.0             4/0/0
                                   LDP.iso.0              0/0/0
                                   LDP.mpls.0             0/0/0
                                   LDP.inet6.0            0/0/0
                                   LDP.l2circuit.0        0/0/0
```

```
OSPF                vrf
                               OSPF.inet.0            7/0/0
                               OSPF.iso.0             0/0/0
                               OSPF.inet6.0           0/0/0
RIP                 vrf
                               RIP.inet.0             6/0/0
                               RIP.iso.0              0/0/0
                               RIP.inet6.0            0/0/0
STATIC              vrf
                               STATIC.inet.0          4/0/0
                               STATIC.iso.0           0/0/0
                               STATIC.inet6.0         0/0/0
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Example: Transporting IPv6 Traffic Across IPv4 Using Filter-Based Tunneling

Example: Configuring the Helper Capability Mode for OSPFv3 Graceful Restart

show route next-hop

IN THIS SECTION

Syntax

```
show route next-hop next-hop
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route next-hop next-hop
<brief | detail | extensive | terse>
```

Description

Display the entries in the routing table that are being sent to the specified next-hop address.

Options

brief | detail | extensive | terse	(Optional) Display the specified level of ouput.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.
next-hop	Next-hop address.

Required Privilege Level

view

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 next-hop

```
user@host> show route next-hop 192.168.71.254

inet.0: 18 destinations, 18 routes (17 active, 0 holddown, 1 hidden)
Restart Complete
+ = Active Route, - = Last Active, * = Both

10.10.0.0/16       *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
10.209.0.0/16      *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
172.16.0.0/12      *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
192.168.0.0/16     *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
192.168.102.0/23   *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
207.17.136.0/24    *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0
207.17.136.192/32  *[Static/5] 06:26:25
                    > to 192.168.71.254 via fxp0.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

red.inet.0: 4 destinations, 5 routes (4 active, 0 holddown, 0 hidden)
```

```
Restart Complete

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

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 next-hop terse

```
user@host> show route next-hop 192.168.71.254 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
* 10.10.0.0/16       S   5                       >192.168.71.254
* 10.209.0.0/16      S   5                       >192.168.71.254
* 172.16.0.0/12      S   5                       >192.168.71.254
* 192.168.0.0/16     S   5                       >192.168.71.254
* 192.168.102.0/23   S   5                       >192.168.71.254
* 207.17.136.0/24    S   5                       >192.168.71.254
* 207.17.136.192/32  S   5                       >192.168.71.254

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

red.inet.0: 4 destinations, 5 routes (4 active, 0 holddown, 0 hidden)
Restart Complete

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
```

```
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)
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route output

IN THIS SECTION

Syntax

```
show route output (address ip-address | interface interface-name)
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route output (address ip-address | interface interface-name)
<brief | detail | extensive | terse>
```

Description

Display the entries in the routing table learned through static routes and interior gateway protocols that are to be sent out the interface with either the specified IP address or specified name.

To view routes advertised to a neighbor or received from a neighbor for the BGP protocol, use the `show route advertising-protocol bgp` and `show route receive-protocol bgp` commands instead.

Options

address *ip-address*	Display entries in the routing table that are to be sent out the interface with the specified IP address.
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`.
interface *interface-name*	Display entries in the routing table that are to be sent out the interface with the specified name.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

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 output address

```
user@host> show route output address 172.16.36.1/24

inet.0: 28 destinations, 30 routes (27 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.36.0/24        *[Direct/0] 00:19:56
                    > via so-0/1/2.0
                    [OSPF/10] 00:19:55, metric 1
                    > via so-0/1/2.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

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

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

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

private1__.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

show route output address detail

```
user@host> show route output address 172.16.36.1 detail

inet.0: 28 destinations, 30 routes (27 active, 0 holddown, 1 hidden)
172.16.36.0/24 (2 entries, 0 announced)
        *Direct Preference: 0
                Next hop type: Interface
                Next-hop reference count: 1
                Next hop: via so-0/1/2.0, selected
                State: <Active Int>
                Age: 23:00
                Task: IF
                AS path: I
         OSPF   Preference: 10
                Next-hop reference count: 1
                Next hop: via so-0/1/2.0, selected
                State: <Int>
                Inactive reason: Route Preference
                Age: 22:59      Metric: 1
                Area: 0.0.0.0
                Task: OSPF
                AS path: I

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

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

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

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

private1__.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

show route output address extensive

The output for the `show route output address extensive` command is identical to that of the `show route output address detail` command. For sample output, see "show route output address detail" on page 752.

show route output address terse

```
user@host> show route output address 172.16.36.1 terse

inet.0: 28 destinations, 30 routes (27 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 172.16.36.0/24        D   0                       >so-0/1/2.0
                     O  10          1            >so-0/1/2.0

private1__.inet.0: 2 destinations, 3 routes (2 active, 0 holddown, 0 hidden)

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

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

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

private1__.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route protocol

IN THIS SECTION

Syntax

```
show route protocol protocol
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route protocol protocol
<brief | detail | extensive | terse>
```

Syntax (PTX and ACX Series Switches)

```
show route protocol protocol
(index index | prefix prefix)
<brief | detail | extensive | terse>
```

Operational mode

`cli` mode

`cli-pfe` mode.

Description

Display the route entries in the routing table that were learned from a particular protocol. Displays details about routes installed in `cli-pfe` mode after FIB compression of FIB compression statistics (PTX and ACX series devices only).

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

protocol — Protocol from which the route was learned:

- `access`—Access route for use by DHCP application
- `access-internal`—Access-internal route for use by DHCP application
- `aggregate`—Locally generated aggregate route
- `arp`—Route learned through the Address Resolution Protocol

- `atmvpn`—Asynchronous Transfer Mode virtual private network
- `bgp`—Border Gateway Protocol
- `ccc`—Circuit cross-connect
- `direct`—Directly connected route
- `dvmrp`—Distance Vector Multicast Routing Protocol
- `esis`—End System-to-Intermediate System
- `flow`—Locally defined flow-specification route
- `frr`—Precomputed protection route or backup route used when a link goes down
- `isis`—Intermediate System-to-Intermediate System
- `ldp`—Label Distribution Protocol
- `l2circuit`—Layer 2 circuit
- `l2vpn`—Layer 2 virtual private network
- `local`—Local address
- `mpls`—Multiprotocol Label Switching
- `msdp`—Multicast Source Discovery Protocol
- `ospf`—Open Shortest Path First versions 2 and 3
- `ospf2`—Open Shortest Path First versions 2 only
- `ospf3`—Open Shortest Path First version 3 only
- `pim`—Protocol Independent Multicast
- `rip`—Routing Information Protocol
- `ripng`—Routing Information Protocol next generation
- `rsvp`—Resource Reservation Protocol
- `rtarget`—Local route target virtual private network
- `static`—Statically defined route

- `tunnel`—Dynamic tunnel
- `vpn`—Virtual private network

> **NOTE**: EX Series switches run a subset of these protocols. See the switch CLI for details.

Required Privilege Level

view

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 protocol access

```
user@host> show route protocol access
inet.0: 30380 destinations, 30382 routes (30379 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

13.160.0.3/32      *[Access/13] 00:00:09
                    > to 13.160.0.2 via fe-0/0/0.0
13.160.0.4/32      *[Access/13] 00:00:09
                    > to 13.160.0.2 via fe-0/0/0.0
13.160.0.5/32      *[Access/13] 00:00:09
                    > to 13.160.0.2 via fe-0/0/0.0
```

show route protocol arp

```
user@host>  show route protocol arp
inet.0: 43 destinations, 43 routes (42 active, 0 holddown, 1 hidden)

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

cust1.inet.0: 1033 destinations, 2043 routes (1033 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

20.20.1.3/32        [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.4/32        [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.5/32        [ARP/4294967293] 00:04:32, from 20.20.1.1
                      Unusable
20.20.1.6/32        [ARP/4294967293] 00:04:34, from 20.20.1.1
                      Unusable
20.20.1.7/32        [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.8/32        [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.9/32        [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.10/32       [ARP/4294967293] 00:04:35, from 20.20.1.1
                      Unusable
20.20.1.11/32       [ARP/4294967293] 00:04:33, from 20.20.1.1
                      Unusable
20.20.1.12/32       [ARP/4294967293] 00:04:33, from 20.20.1.1
                      Unusable
20.20.1.13/32       [ARP/4294967293] 00:04:33, from 20.20.1.1
                      Unusable
...
```

show route protocol bgp

```
user@host> show route protocol bgp 192.168.64.0/21
inet.0: 335832 destinations, 335833 routes (335383 active, 0 holddown, 450 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
192.168.64.0/21       *[BGP/170] 6d 10:41:16, localpref 100, from 192.168.69.71
                      AS path: 10458 14203 2914 4788 4788 I
                    > to 192.168.167.254 via fxp0.0
```

show route protocol direct

```
user@host> show route protocol direct

inet.0: 335843 destinations, 335844 routes (335394 active, 0 holddown, 450 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.8.0/24         *[Direct/0] 17w0d 10:31:49
                    > via fe-1/3/1.0
10.255.165.1/32    *[Direct/0] 25w4d 04:13:18
                    > via lo0.0
172.16.30.0/24      *[Direct/0] 17w0d 23:06:26
                    > via fe-1/3/2.0
192.168.164.0/22   *[Direct/0] 25w4d 04:13:20
                    > via fxp0.0

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

47.0005.80ff.f800.0000.0108.0001.0102.5516.5001/152
                   *[Direct/0] 25w4d 04:13:21
                    > via lo0.0

inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::10:255:165:1/128
                   *[Direct/0] 25w4d 04:13:21
                    > via lo0.0
fe80::2a0:a5ff:fe12:ad7/128
                   *[Direct/0] 25w4d 04:13:21
                    > via lo0.0
```

show route protocol frr

```
user@host> show route protocol frr
inet.0: 43 destinations, 43 routes (42 active, 0 holddown, 1 hidden)

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

cust1.inet.0: 1033 destinations, 2043 routes (1033 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

20.20.1.3/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.3 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.4/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.4 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.5/32       *[FRR/200] 00:05:35, from 20.20.1.1
                    > to 20.20.1.5 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.6/32       *[FRR/200] 00:05:37, from 20.20.1.1
                    > to 20.20.1.6 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.7/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.7 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.8/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.8 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.9/32       *[FRR/200] 00:05:38, from 20.20.1.1
                    > to 20.20.1.9 via ge-4/1/0.0
                      to 10.10.15.1 via ge-0/2/4.0, Push 16, Push 299792(top)
20.20.1.10/32      *[FRR/200] 00:05:38, from 20.20.1.1
...
```

show route protocol ldp

```
user@host> show route protocol ldp
inet.0: 12 destinations, 13 routes (12 active, 0 holddown, 0 hidden)

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

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

192.168.16.1/32    *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0, Push 100000
192.168.17.1/32    *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0

private1__.inet.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

mpls.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

100064             *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0, Pop
100064(S=0)        *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0, Pop
100080             *[LDP/9] 1d 23:03:35, metric 1
                    > via t1-4/0/0.0, Swap 100000
```

show route protocol ospf (Layer 3 VPN)

```
user@host> show route protocol ospf
inet.0: 40 destinations, 40 routes (39 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

10.39.1.4/30       *[OSPF/10] 00:05:18, metric 4
                    > via t3-3/2/0.0
10.39.1.8/30        [OSPF/10] 00:05:18, metric 2
                    > via t3-3/2/0.0
10.255.14.171/32   *[OSPF/10] 00:05:18, metric 4
                    > via t3-3/2/0.0
10.255.14.179/32   *[OSPF/10] 00:05:18, metric 2
                    > via t3-3/2/0.0
172.16.233.5/32       *[OSPF/10] 20:25:55, metric 1

VPN-AB.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.39.1.16/30       [OSPF/10] 00:05:43, metric 1
                    > via so-0/2/2.0
10.255.14.173/32   *[OSPF/10] 00:05:43, metric 1
```

```
                    > via so-0/2/2.0
172.16.233.5/32       *[OSPF/10] 20:26:20, metric 1
```

show route protocol rip

```
user@host> show route protocol rip
inet.0: 26 destinations, 27 routes (25 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

VPN-AB.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
10.255.14.177/32   *[RIP/100] 20:24:34, metric 2
                    > to 10.39.1.22 via t3-0/2/2.0
172.16.233.9/32       *[RIP/100] 00:03:59, metric 1
```

show route protocol rsvp extensive

```
user@host> show route protocol rsvp extensive
inet.0: 37 destinations, 37 routes (36 active, 0 holddown, 1 hidden)

inet.3: 3 destinations, 4 routes (3 active, 0 holddown, 0 hidden)
192.168.4.4/32 (2 entries, 1 announced)
        State: <FlashAll>
        *RSVP   Preference: 7/1
                Next hop type: Router, Next hop index: 0
                Address: 0xaad4fbc
                Next-hop reference count: 2, key opaque handle: 0x0, non-key opaque handle: 0x0
                Next hop: 192.168.0.1 via ge-0/0/0.0 weight 0x1, selected
                Label-switched-path PE1_to_PE2_1
                Label operation: Push 299920
                Label TTL action: prop-ttl
                Load balance label: Label 299920: None;
                Label element ptr: 0x9938be8
                Label parent element ptr: 0x0
                Label element references: 3
                Label element child references: 1
                Label element lsp id: 2
                Session Id: 0
                Next hop: 192.168.1.1 via ge-0/0/1.0 weight 0x8001 uflags Mark connection
protection tlv
```

```
                  Label-switched-path Bypass->192.168.0.1
                  Label operation: Push 299920
                  Label TTL action: prop-ttl
                  Load balance label: Label 299920: None;
                  Label element ptr: 0x9938eb8
                  Label parent element ptr: 0x9938be8
                  Label element references: 2
                  Label element child references: 0
                  Label element lsp id: 2
                  Session Id: 0
                  State: <Active Int>
                  Local AS:   100
                  Age: 1:20       Metric: 100
                  Validation State: unverified
                  Actual IGP metric: 2
                  Task: RSVP
                  Announcement bits (2): 0-Resolve tree 3 3-Resolve tree 1
                  AS path: I
                  Thread: junos-main
```

show route protocol index prefix detail (ACX7100-32C, ACX7100-48L, and ACX7509 Devices)

Displays if a route is installed or not when FIB compression is enabledIf the route is not installed, the command displays the details of the parent route installed, that compressed the route.

```
user@root:pfe> show route proto 2 index 0 prefix 200.1.0.16 detail
Protocol: 2
Protocol     : IPv4
Table        : default
Prefix       : 200.1.0.16 (primary)
NH           : 25035 (software)
Flags        : 0x00008000
Details      :
guid          : 889058232562
type          :  user
nhid          :  25035
Forwarding state:
installed?    :  no
(Installed parent: 200.1.0/23)
```

Release Information

Command introduced before Junos OS Release 7.4.

`ospf2` and `ospf3` options introduced in Junos OS Release 9.2.

`ospf2` and `ospf3` options introduced in Junos OS Release 9.2 for EX Series switches.

`flow` option introduced in Junos OS Release 10.0.

`flow` option introduced in Junos OS Release 10.0 for EX Series switches.

RELATED DOCUMENTATION

show route

show route detail

show route extensive

show route terse

show route receive-protocol

IN THIS SECTION

Syntax

```
show route receive-protocol protocol neighbor-address
 <brief | detail | extensive | terse>
<logical-system (all | logical-system-name)
```

Syntax (EX Series Switches)

```
show route receive-protocol protocol neighbor-address
<brief | detail | extensive | terse>
```

Description

Display the routing information as it was received through a particular neighbor using a particular dynamic routing protocol.

Options

brief | detail | extensive | terse	(Optional) Display the specified level of output.
logical-system (all | ***logical-system-name*****)**	(Optional) Perform this operation on all logical systems or on a particular logical system.
protocol neighbor-address	Protocol transmitting the route (`bgp`, `dvmrp`, `msdp`, `pim`, `rip`, or `ripng`) and address of the neighboring router from which the route entry was received.

Additional Information

The output displays the selected routes and the attributes with which they were received, but does not show the effects of import policy on the routing attributes.

Required Privilege Level

view

Output Fields

Table 44 on page 766 describes the output fields for the `show route receive-protocol` command. Output fields are listed in the approximate order in which they appear.

Table 44: show route receive-protocol Output Fields

Field Name	Field Description	Level of Output
routing-table-name	Name of the routing table—for example, inet.0.	All levels
number destinations	Number of destinations for which there are routes in the routing table.	All levels
number routes	Number of routes in the routing table and total number of routes in the following states: • `active` • `holddown` (routes that are in pending state before being declared inactive) • `hidden` (routes that are not used because of a routing policy)	All levels
Prefix	Destination prefix.	none `brief`
MED	Multiple exit discriminator value included in the route.	none `brief`

Table 44: show route receive-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
destination-prefix (entry, announced)	Destination prefix. 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.	`detail extensive`
Accepted LongLivedStale	The LongLivedStale flag indicates that the route was marked LLGR-stale by this router, as part of the operation of LLGR receiver mode. Either this flag or the LongLivedStaleImport flag may be displayed for a route. Neither of these flags are displayed at the same time as the Stale (ordinary GR stale) flag.	`detail extensive`
Accepted LongLivedStaleImport	The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy. Either this flag or the LongLivedStale flag may be displayed for a route. Neither of these flags are displayed at the same time as the Stale (ordinary GR stale) flag. Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and import into the inet.0 routing table	`detail extensive`
ImportAccepted LongLivedStaleImport	Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and imported into the inet.0 routing table The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy.	`detail extensive`
Route Distinguisher	64-bit prefix added to IP subnets to make them unique.	`detail extensive`
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.	`detail extensive`
VPN Label	Virtual private network (VPN) label. Packets are sent between CE and PE routing devices by advertising VPN labels. VPN labels transit over either an RSVP or an LDP label-switched path (LSP) tunnel.	`detail extensive`

Table 44: show route receive-protocol Output Fields *(Continued)*

Field Name	Field Description	Level of Output
Next hop	Next hop to the destination. An angle bracket (>) indicates that the route is the selected route.	All levels
Localpref or Lclpref	Local preference value included in the route.	All levels
AS path	Autonomous system (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: • `I`—IGP. • `E`—EGP. • `?`—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • `[ ]`—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 the AS-path merge process, as defined in RFC 4893. • `[ ]`—If more than one AS number is configured on the router, 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. **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.	All levels

Table 44: show route receive-protocol Output Fields ***(Continued)***

Field Name	Field Description	Level of Output
Route Labels	Stack of labels carried in the BGP route update.	`detail extensive`
Cluster list	(For route reflected output only) Cluster ID sent by the route reflector.	`detail extensive`
Originator ID	(For route reflected output only) Address of routing device that originally sent the route to the route reflector.	`detail extensive`
Communities	Community path attribute for the route.	`detail extensive`
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.	`detail extensive`
Attrset AS	Number, local preference, and path of the AS that originated the route. These values are stored in the `Attrset` attribute at the originating routing device.	`detail extensive`
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).	`detail extensive`
control flags	Control flags: `none` or `Site Down`.	`detail extensive`
mtu	Maximum transmission unit (MTU) of the Layer 2 circuit.	`detail extensive`

Sample Output

show route receive-protocol bgp

```
user@host> show route receive-protocol bgp 10.255.245.215

inet.0: 28 destinations, 33 routes (27 active, 0 holddown, 1 hidden)
Prefix                  Next hop              MED     Lclpref    AS path
```

```
10.22.1.0/24            10.255.245.215      0       100        I
10.22.2.0/24            10.255.245.215      0       100        I
```

show route receive-protocol bgp detail (BGP Multicast)

```
user@host> show route receive-protocol bgp 10.4.6.1 detail
bgpmcast.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
* 4:3:0:0:32:10.2.20.20:32:225.1.1.1:10.4.4.4:10.4.0.1/240 (1 entry, 1 announced)
     Accepted
     Nexthop: 10.4.0.1
     Localpref: 100
     AS path: 65002 I
     Communities: target:10.4.4.4:0
     Tunnel type: AnyEncap, RPF tunnel:, Remote end point: 10.1.4.4
     Tunnel type: AnyEncap, Remote end point: 10.2.4.4
     Tunnel type: AnyEncap, Remote end point: 10.3.4.4
     Tunnel type: AnyEncap, Remote end point: 10.4.6.4
     Tunnel type: AnyEncap, Remote end point: 10.4.20.4
```

Show route receive protocol (Segment Routing Traffic Engineering)

```
show route receive protocol bgp 10.1.1.4
bgp.inetcolor.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

* 50-10.4.4.4-1234<sr6>/96 (1 entry, 0 announced)
     Import Accepted
     Distinguisher: 50
     Color: 1234
     Nexthop: 10.1.1.4
     Localpref: 100
     AS path: 65003 I
     Communities: target:10.1.1.1:1


inetcolor.0: 6 destinations, 7 routes (6 active, 0 holddown, 0 hidden)
* 10.4.4.4-1234<c6>/64 (1 entry, 1 announced)
     Import Accepted
     Color: 1234
     Nexthop: 10.1.1.4
     Localpref: 100
```

```
     AS path: 65003 I
     Communities: target:10.1.1.1:1


user@host# run show route receive-protocol bgp 2001:db8:5001:1::4
bgp.inet6color.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

* 50-2001:db8:1::4-1234<sr6>/192 (1 entry, 0 announced)
     Import Accepted
     Distinguisher: 50
     Color: 1234
     Nexthop: ::ffff:10.1.1.4
     Localpref: 100
     AS path: 65003 I
     Communities: target:10.1.1.1:1


inet6color.0: 6 destinations, 7 routes (6 active, 0 holddown, 0 hidden)
* 2001:db8::5-1234<c6>/160 (1 entry, 1 announced)
     Import Accepted
     Color: 1234
     Nexthop: ::ffff:10.1.1.5
     Localpref: 100
     AS path: 65003 I
     Communities: target:10.1.1.1:1
```

Release Information

Command introduced before Junos OS Release 7.4.

show route table

IN THIS SECTION

Syntax

```
show route table routing-table-name
<brief | detail | extensive | terse>
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches, QFX Series Switches)

```
show route table routing-table-name
<brief | detail | extensive | terse>
```

Description

Display the route entries in a particular routing table.

Options

brief | detail | extensive | terse	(Optional) Display the specified level of output.

logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system. This option is only supported on Junos OS.
routing-table-name	Display route entries for all routing tables whose names begin with this string (for example, inet.0 and inet6.0 are both displayed when you run the `show route table inet` command).

Required Privilege Level

view

Output Fields

Table 45 on page 773 describes the output fields for the `show route table` command. Output fields are listed in the approximate order in which they appear.

Table 45: show route table Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).

Table 45: show route table Output Fields *(Continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th></tr>
<tr><td>Restart complete</td><td>All protocols have restarted for this routing table.

Restart state:
<code>Pending:</code>protocol-name—List of protocols that have not yet completed graceful restart for this routing table.
<code>Complete</code>—All protocols have restarted for this routing table.
For example, if the output shows-
<pre>LDP.inet.0   : 5 routes (4 active, 1 holddown, 0 hidden)
    Restart Pending: OSPF LDP VPN</pre>This indicates that <code>OSPF</code>, <code>LDP</code>, and <code>VPN</code> protocols did not restart for the <code>LDP.inet.0</code> routing table.
<pre>vpls_1.l2vpn.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
Restart Complete</pre>This indicates that all protocols have restarted for the <code>vpls_1.l2vpn.0</code> routing table.</td></tr>
<tr><td>number destinations</td><td>Number of destinations for which there are routes in the routing table.</td></tr>
<tr><td>number routes</td><td>Number of routes in the routing table and total number of routes in the following states:
<code>active</code> (routes that are active)
<code>holddown</code> (routes that are in the pending state before being declared inactive)
<code>hidden</code> (routes that are not used because of a routing policy)</td></tr>
</table>

Table 45: show route table Output Fields *(Continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th></tr>
<tr><td>route-destination (entry, announced)</td><td>Route destination (for example:10.0.0.1/24). The <code>entry</code> value is the number of routes for this destination, and the <code>announced</code> value is the number of routes being announced for this destination. Sometimes the route destination is presented in another format, such as:

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).

neighbor-address—Address of the neighbor.
control-word-status—Whether the use of the control word has been negotiated for this virtual circuit: <code>NoCtrlWord</code> or <code>CtrlWord</code>.
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.
<code>vc-id</code>—Virtual circuit identifier.
source—Source of the advertisement: <code>Local</code> or <code>Remote</code>.

<code>inclusive multicast Ethernet tag route</code>—Type of route destination represented by (for example, 3:100.100.100.10:100::0::10::100.100.100.10/384):

route distinguisher—(8 octets) Route distinguisher (RD) must be the RD of the EVPN instance (EVI) that is advertising the NLRI.
Ethernet tag ID—(4 octets) Identifier of the Ethernet tag. Can set to 0 or to a valid Ethernet tag value.
IP address length—(1 octet) Length of IP address in bits.
originating router’s IP address—(4 or 16 octets) Must set to the provider edge (PE) device’s IP address. This address should be common for all EVIs on the PE device, and may be the PE device's loopback address.

</td></tr>
</table>

Table 45: show route table Output Fields *(Continued)*

Field Name	Field Description
label stacking	(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. • `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).
[*protocol*, *preference*]	Protocol from which the route was learned and the preference value for the route. • `+`—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. 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.
Level	(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.
Route Distinguisher	IP subnet augmented with a 64-bit prefix.
PMSI	Provider multicast service interface (MVPN routing table).

Table 45: show route table Output Fields *(Continued)*

Field Name	Field Description
Next-hop type	Type of next hop. For a description of possible values for this field, see Table 46 on page 782.
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	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: • 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).

Table 45: show route table Output Fields *(Continued)*

Field Name	Field Description
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 47 on page 784.
Local AS	AS number of the local routing devices.
Age	How long the route has been known.
AIGP	Accumulated interior gateway protocol (AIGP) BGP attribute.
Metric*n*	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.
MED-plus-IGP	Metric value for BGP path selection to which the IGP cost to the next-hop destination has been added.
TTL-Action	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.
Task	Name of the protocol that has added the route.

Table 45: show route table Output Fields *(Continued)*

Field Name	Field Description
Announcement bits	The number of BGP peers or protocols to which Junos OS has announced this route, followed by the list of the recipients of the announcement. Junos OS can also announce the route to the kernel routing table (KRT) for installing the route into the Packet Forwarding Engine, to a resolve tree, a Layer 2 VC, or even a VPN. For example, *n*-Resolve inet indicates that the specified route is used for route resolution for next hops found in the routing table. • *n*—An index used by Juniper Networks customer support only.
AS path	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: • I—IGP. • E—EGP. • Recorded—The AS path is recorded by the sample process (sampled). • ?—Incomplete; typically, the AS path was aggregated. When AS path numbers are included in the route, the format is as follows: • [ ]—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. **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.

Table 45: show route table Output Fields *(Continued)*

Field Name	Field Description
validation-state	(BGP-learned routes) Validation status of the route: • `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. • `Unverified`—Indicates that the origin of the prefix is not verified against the database. This is because the database got populated and the validation is not called for in the BGP import policy, although origin validation is enabled, or the origin validation is not enabled for the BGP peers. • `Valid`—Indicates that the prefix and autonomous system pair are found in the database.
FECs bound to route	Indicates point-to-multipoint root address, multicast source address, and multicast group address when multipoint LDP (M-LDP) inband signaling is configured.
Primary Upstream	When multipoint LDP with multicast-only fast reroute (MoFRR) is configured, indicates the primary upstream path. MoFRR transmits a multicast join message from a receiver toward a source on a primary path, while also transmitting a secondary multicast join message from the receiver toward the source on a backup path.
RPF Nexthops	When multipoint LDP with MoFRR is configured, indicates the reverse-path forwarding (RPF) next-hop information. Data packets are received from both the primary path and the secondary paths. The redundant packets are discarded at topology merge points due to the RPF checks.
Label	Multiple MPLS labels are used to control MoFRR stream selection. Each label represents a separate route, but each references the same interface list check. Only the primary label is forwarded while all others are dropped. Multiple interfaces can receive packets using the same label.
weight	Value used to distinguish MoFRR primary and backup routes. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible.

Table 45: show route table Output Fields *(Continued)*

Field Name	Field Description
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 48 on page 788 for all possible values for this field.
Layer2-info: encaps	Layer 2 encapsulation (for example, VPLS).
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 LongLivedStale	The LongLivedStale flag indicates that the route was marked LLGR-stale by this router, as part of the operation of LLGR receiver mode. Either this flag or the LongLivedStaleImport flag might be displayed for a route. Neither of these flags is displayed at the same time as the Stale (ordinary GR stale) flag.

Table 45: show route table Output Fields *(Continued)*

Field Name	Field Description
Accepted LongLivedStaleImport	The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy. Either this flag or the LongLivedStale flag might be displayed for a route. Neither of these flags is displayed at the same time as the Stale (ordinary GR stale) flag. Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and import into the inet.0 routing table
ImportAccepted LongLivedStaleImport	Accept all received BGP long-lived graceful restart (LLGR) and LLGR stale routes learned from configured neighbors and imported into the inet.0 routing table The LongLivedStaleImport flag indicates that the route was marked LLGR-stale when it was received from a peer, or by import policy.
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 46 on page 782 describes all possible values for the Next-hop Types output field.

Table 46: Next-hop Types Output Field Values

Next-Hop Type	Description
Broadcast (bcast)	Broadcast next hop.

Table 46: Next-hop Types Output Field Values *(Continued)*

Next-Hop Type	Description
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.
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.

Table 46: Next-hop Types Output Field Values *(Continued)*

Next-Hop Type	Description
Reject (rjct)	Discard. An ICMP unreachable message was sent.
Resolve (rslv)	Resolving next hop.
Routed multicast (mcrt)	Regular multicast next hop.
Router	A specific node or set of nodes to which the routing device forwards packets that match the route prefix. To qualify as a next-hop type router, the route must meet the following criteria: • 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 47 on page 784 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 47: State Output Field Values

Value	Description
Accounting	Route needs accounting.
Active	Route is active.

Table 47: State Output Field Values *(Continued)*

Value	Description
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.

Table 47: State Output Field Values *(Continued)*

Value	Description
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).
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.

Table 47: State Output Field Values *(Continued)*

Value	Description
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: • 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.
VxlanLocalRT	Route is an EVPN Type 5 route (IP prefix route).

Table 48 on page 788 describes the possible values for the Communities output field.

Table 48: Communities Output Field Values

Value	Description
area-number	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.
bandwidth: *local AS number:link-bandwidth-number*	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.
domain-id	Unique configurable number that identifies the OSPF domain.
GBP Tag	Assigned Group Based Policy (GBP) tag from 1 through 65535.
domain-id-vendor	Unique configurable number that further identifies the OSPF domain.
link-bandwidth-number	Link-bandwidth number: from 0 through 4,294,967,295 (bytes per second).
local AS number	Local AS number: from 1 through 65,535.
options	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.
origin	(Used with VPNs) Identifies where the route came from.
ospf-route-type	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 be0); 7 for NSSA routes; or 129 for sham link endpoint addresses.
route-type-vendor	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*.

Table 48: Communities Output Field Values *(Continued)*

Value	Description
rte-type	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*.
target	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.
unknown IANA	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.
unknown OSPF vendor community	Incoming IANA codes with a value above 0x8000. This code of the BGP extended community attribute is accepted, but it is not recognized.
evpn-mcast-flags	Identifies the value in the multicast flags extended community, which includes bits that indicate whether snooping or optimized intersubnet multicast (OISM) is enabled. For example: • 0x01—IGMP snooping bit • 0x02—MLD snooping bit • 0x08—OISM bit
evpn-l2-info	Identifies whether Multihomed Proxy MAC and IP Address Route Advertisement is enabled. A value of 0x20 indicates that the proxy bit is set. . Use the `show bridge mac-ip-table extensive` statement to determine whether the MAC and IP address route was learned locally or from a PE device.

Sample Output

show route table bgp.l2vpn.0

```
user@host> show route table bgp.l2vpn.0
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 evpn-vs2.evpn.0 extensive

```
user@host> show route table evpn-vs2.evpn.0 extensive
evpn-vs2.evpn.0: 30 destinations, 30 routes (30 active, 0 holddown, 0 hidden)
Restart Complete
2:10.3.3.3:2::601: :2c:6b:f5:f2:87:f0/304 MAC/IP (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 10.3.3.3:2
                Next hop type: Indirect, Next hop index: 0
                Address: 0x76b66dc
                Next-hop reference count: 32
                Source: 10.2.2.2
                Protocol next hop: 10.3.3.3
                Indirect next hop: 0x2 no-forward INH Session ID: 0
                State: <Secondary Active Ext>
                Local AS:   102 Peer AS:   201
                Age: 22:56:12     Metric2: 1
                Validation State: unverified
                Task: BGP_201.10.2.2.2
                Announcement bits (1): 0-evpn-vs2-evpn
                AS path: 201 203 I
                Communities: target:5:5 encapsulation:vxlan(0x8) evpn-default-gateway gbp-tag:
200
```

show route table bgpmcast.0 extensive

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


4:3:0:0:32:10.2.20.20:32:225.1.1.1:10.4.4.4:10.20.20.20/240 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Tunnel type: AnyEncap, RPF tunnel:, Remote end point: 10.1.4.4
                Tunnel type: AnyEncap, Remote end point: 10.2.4.4
                Tunnel type: AnyEncap, Remote end point: 10.3.4.4
                Tunnel type: AnyEncap, Remote end point: 10.4.6.4
                Next hop type: Indirect, Next hop index: 0
                Address: 0xc54639c
                Next-hop reference count: 1
                Source: 10.1.1.1
                Protocol next hop: 10.20.20.20
                Indirect next hop: 0x2 no-forward INH Session ID: 0x0
                State: <Active Int Ext>
                Local AS:   65100 Peer AS:   65100
                Age: 30         Metric2: 0
                Validation State: unverified
                Task: BGP_100.1.1.1.1
                Announcement bits (1): 0-bgpmcast global task
                AS path: I
                Communities: target:10.4.4.4:0
                Accepted
                Localpref: 100
                Router ID: 10.1.1.1
                Indirect next hops: 1
                        Protocol next hop: 10.20.20.20
                        Indirect next hop: 0x2 no-forward INH Session ID: 0x0
                        Indirect path forwarding next hops: 1
                                Next hop type: Router
                                Next hop: 10.4.20.20 via ge-0/0/5.0
                                Session Id: 0x140
                                10.20.20.20/32 Originating RIB: inet.0
                                  Node path count: 1
                                  Forwarding nexthops: 1
                                        Next hop type: Router
```

```
                                          Next hop: 10.4.20.20 via ge-0/0/5.0
                                          Session Id: 0x140
```

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 172.16.5.254 via fxp0.0
10.0.0.1/32         *[Direct/0] 00:51:58
                    > via at-5/3/0.0
10.0.0.2/32         *[Local/0] 00:51:58
                     Local
10.12.12.21/32     *[Local/0] 00:51:57
                     Reject
10.13.13.13/32     *[Direct/0] 00:51:58
                    > via t3-5/2/1.0
10.13.13.14/32     *[Local/0] 00:51:58
                     Local
10.13.13.21/32     *[Local/0] 00:51:58
                     Local
10.13.13.22/32     *[Direct/0] 00:33:59
                    > via t3-5/2/0.0
127.0.0.1/32        [Direct/0] 00:51:58
                    > via lo0.0
10.222.5.0/24     *[Direct/0] 00:51:58
                    > via fxp0.0
10.222.5.81/32    *[Local/0] 00:51:58
                     Local
```

show route table inet.3

```
user@host> show route table inet.3
inet.3: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32        *[LDP/9] 00:25:43, metric 10, tag 200
```

```
                   to 10.2.94.2 via lt-1/2/0.49
                 > to 10.2.3.2 via lt-1/2/0.23
```

show route table inet.3 protocol ospf

```
user@host> show route table inet.3 protocol ospf
inet.3: 9 destinations, 18 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1.1.1.20/32         [L-OSPF/10] 1d 00:00:56, metric 2
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800020
                      to 10.0.6.60 via lt-1/2/0.12, Push 800020, Push 800030(top)
1.1.1.30/32         [L-OSPF/10] 1d 00:01:01, metric 3
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800030
                      to 10.0.6.60 via lt-1/2/0.12, Push 800030
1.1.1.40/32         [L-OSPF/10] 1d 00:01:01, metric 4
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800040
                      to 10.0.6.60 via lt-1/2/0.12, Push 800040
1.1.1.50/32         [L-OSPF/10] 1d 00:01:01, metric 5
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800050
                      to 10.0.6.60 via lt-1/2/0.12, Push 800050
1.1.1.60/32         [L-OSPF/10] 1d 00:01:01, metric 6
                    > to 10.0.10.70 via lt-1/2/0.14, Push 800060
                      to 10.0.6.60 via lt-1/2/0.12, Pop
```

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 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 lsdist.0

```
user@host> show route table lsdist.0

lsdist.0: 21 destinations, 21 routes (21 active, 0 holddown, 0 hidden)
```

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

NODE { AS:100 ISO:0003.0003.0303.00 ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
NODE { AS:100 ISO:1282.0404.9202.00 ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
NODE { AS:100 ISO:1282.0404.9202.02 ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
LINK { Local { AS:100 ISO:0003.0003.0303.00 }.{ IPv4:23.0.0.2 IPv6:23::2 } Remote { AS:100
ISO:1282.0404.9202.02 }.{ } ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
LINK { Local { AS:100 ISO:1282.0404.9202.00 }.{ IPv4:23.0.0.1 IPv6:23::1 } Remote { AS:100
ISO:1282.0404.9202.02 }.{ } ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
LINK { Local { AS:100 ISO:1282.0404.9202.02 }.{ } Remote { AS:100 ISO:0003.0003.0303.00 }.{ }
ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
LINK { Local { AS:100 ISO:1282.0404.9202.02 }.{ } Remote { AS:100 ISO:1282.0404.9202.00 }.{ }
ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:3.3.3.3/32 } ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:32.32.32.32/32 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:33.33.33.33/32 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:100.100.100.0/32 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv4:128.204.49.193/32 } ISIS-L2:0 }/
```

```
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv4:2.2.2.2/32 } ISIS-L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv4:128.204.49.202/32 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv6:3::3/128 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv6:23::/64 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv6:100:100:100::100/128 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:0003.0003.0303.00 } { IPv6:abcd::128:204:49:193/128 } ISIS-
L2:0 }/1216
                   *[IS-IS/18] 10:19:16
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv6:2::2/128 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv6:23::/64 } ISIS-L2:0 }/
1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
IPV6 PREFIX { Node { AS:100 ISO:1282.0404.9202.00 } { IPv6:abcd::128:204:49:202/128 } ISIS-
L2:0 }/1216
                   *[IS-IS/18] 10:19:07
                       Fictitious
```

show route table lsdist.0 (BGP Confederation Enabled)

```
user@host>show route table lsdist.0 detail
lsdist.0: 72 destinations, 72 routes (72 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

NODE { AS:17 Member-ASN:65002 ISO:0004.0404.0400.00 ISIS-L2:0 }/1216
                   *[BGP/170] 00:00:23, localpref 100
                     AS path: (65002) I, validation-state: unverified
                    >  to 11.14.1.2 via ge-0/0/0.0
NODE { AS:17 Member-ASN:65002 ISO:0005.0505.0500.00 ISIS-L2:0 }/1216
                   *[BGP/170] 00:00:23, localpref 100
                      AS path: (65002) I, validation-state: unverified
                    >  to 11.14.1.2 via ge-0/0/0.0

LINK { Local { AS:17 Member-ASN:65001 ISO:0003.0303.0300.00 }.{ IPv4:13.14.1.1
IPv6:abcd::13:14:1:1 } Remote { AS:17 Member-ASN:65001 ISO:0004.0404.0400.00 }.
{ IPv4:13.14.1.2 } ISIS-L2:0 }/1216
                   *[IS-IS/18] 00:00:48
                       Fictitious
LINK { Local { AS:17 Member-ASN:65001 ISO:0003.0303.0300.00 }.{ IPv4:13.14.2.1
IPv6:abcd::13:14:2:1 } Remote { AS:17 Member-ASN:65001 ISO:0004.0404.0400.00 }.
{ IPv4:13.14.2.2 } ISIS-L2:0 }/1216
                   *[IS-IS/18] 00:00:48
                       Fictitious
```

show route table lsdist.0 detail

```
user@host> show route table lsdist.0 detail
lsdist.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)
NODE { AS:200 ISO:1282.2113.1154.00 ISIS-L1:0 }/1216 (1 entry, 1 announced)
*IS-IS  Preference: 15
Level: 1
Next hop type: Fictitious, Next hop index: 0
Address: 0xc5b3054
Next-hop reference count: 14
…………..
…………..
………….
```

```
Area membership:
47 00 05 80 ff f8 00 00 00 01 08 00 01
SPRING-Capabilities:
- SRGB block [Start: 800000, Range: 4096, Flags: 0xc0]
SPRING-Algorithms:
- Algo: 0
SPRING Flex-Algorithms Definition:
- Flex-Algo: 129
Metric: 0, Calc: 0, priority: 129
- Flags: 0x02, - Inc Any: 0x00040000, - Exclude: 0x00008000, - Inc All: 0x00004000
………….
………….
………….
PREFIX { Node { AS:200 ISO:1282.2113.3158.00 } { IPv4:128.220.13.196/32 } ISIS-L1:0 }/1216 (1
entry, 1 announced)
*IS-IS  Preference: 15
Level: 1
Next hop type: Fictitious, Next hop index: 0
Address: 0xc5b3054
Next-hop reference count: 14
Next hop:
State: <Active NotInstall>
Local AS:   200
Age: 16:16:25
………….
………….
Prefix SID: 10, Flags: 0xe0, Algo: 0
Prefix SID: 780, Flags: 0xe0, Algo: 129
Flex Algo: 129, Flex Algo Metric: 10
………….
………….
IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:4000::/64 } ISIS-L1:0 }/1216 (1
entry, 1 announced)
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 588
                Address: 0x7660f64
                Next-hop reference count: 12, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 21.0.2.1
                Next hop: 21.0.2.1 via ge-0/0/0.0, selected
                Session Id: 320
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 1d 10:20:38
```

```
                Validation State: unverified
                Task: BGP_100.21.0.2.1
                Announcement bits (1): 0-TED Export
                AS path: 100 I
                Accepted
                SRv6 Locator Flags: 0, Algo: 0, Metric: 0
                Localpref: 100
                Router ID: 100.100.100.0
                Thread: junos-main
```

show route table lsdist.0 detail

```
user@host> show route table lsdist.1 detail
SPRING Flex-Algorithms Definition:
                 - Flex-Algo: 128
                   Metric: 2, Calc: 1, Priority: 10
                    - Flags: 0X80000000, - Inc Any: 0x00000002, - Exclude: 0x00000004, - Inc
All: 0x00000002
```

show route table lsdist.0 extensive

The output of the `show route table lsdist.0 extensive` command to include IPv6 Prefix attributes.

```
user@host> show route table lsdist.0 extensive
lsdist.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
NODE { AS:100 ISO:0100.0100.0100.00 ISIS-L1:0 }/1216 (1 entry, 1 announced)
TSI:
LINK-STATE attribute handle 0x75ace70
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 0
                Address: 0x7661124
                Next-hop reference count: 5, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 21.0.2.1
                Next hop: 21.0.2.1 via ge-0/0/0.0, selected
                Session Id: 0
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 17
                Validation State: unverified
```

```
                Task: BGP_100.21.0.2.1
                Announcement bits (1): 0-TED Export
                AS path: 100 I
                Accepted
                IPv4 Router-ids:
                  100.100.100.0
                Area border router: No
                External router: No
                Attached: No
                Overload: No
                Hostname: R0
                Area membership:
                  49 00 05
                SPRING-Algorithms:
                 - Algo: 0
                 - Algo: 1
                SRV6 Capable: - Flags: 0
                SRV6 Node MSD:
                 - Type: 41, Value: 6
                 - Type: 42, Value: 7
                 - Type: 43, Value: 5
                 - Type: 44, Value: 6
                 - Type: 45, Value: 6
                Localpref: 100
                Router ID: 100.100.100.0
                Thread: junos-main

PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv4:100.100.100.0/32 } ISIS-L1:0 }/1216 (1
entry, 1 announced)
TSI:
LINK-STATE attribute handle 0x0
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 0
                Address: 0x7661124
                Next-hop reference count: 5, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 21.0.2.1
                Next hop: 21.0.2.1 via ge-0/0/0.0, selected
                Session Id: 0
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 28
                Validation State: unverified
                Task: BGP_100.21.0.2.1
```

```
                Announcement bits (1): 0-TED Export
                AS path: 100 I
                Accepted
                Localpref: 100
                Router ID: 100.100.100.0
                Thread: junos-main

IPV6 PREFIX { Node { AS:100 ISO:0100.0100.0100.00 } { IPv6:10::10/128 } ISIS-L1:0 }/1216 (1
entry, 1 announced)
TSI:
LINK-STATE attribute handle 0x0
        *BGP    Preference: 170/-101
                Next hop type: Router, Next hop index: 0
                Address: 0x7661124
                Next-hop reference count: 5, key opaque handle: 0x0, non-key opaque handle: 0x0
                Source: 21.0.2.1
                Next hop: 21.0.2.1 via ge-0/0/0.0, selected
                Session Id: 0
                State: <Active Ext>
                Local AS:   200 Peer AS:   100
                Age: 28
                Validation State: unverified
                Task: BGP_100.21.0.2.1
                Announcement bits (1): 0-TED Export
                AS path: 100 I
                Accepted
                Localpref: 100
                Router ID: 100.100.100.0
                Thread: junos-main
```

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.0 protocol ospf

```
user@host> show route table mpls.0 protocol ospf
mpls.0: 29 destinations, 29 routes (29 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

299952             *[L-OSPF/10] 23:59:42, metric 0
                    > to 10.0.10.70 via lt-1/2/0.14, Pop
                      to 10.0.6.60 via lt-1/2/0.12, Swap 800070, Push 800030(top)
299952(S=0)        *[L-OSPF/10] 23:59:42, metric 0
                    > to 10.0.10.70 via lt-1/2/0.14, Pop
                      to 10.0.6.60 via lt-1/2/0.12, Swap 800070, Push 800030(top)
299968             *[L-OSPF/10] 23:59:48, metric 0
                    > to 10.0.6.60 via lt-1/2/0.12, Pop
```

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
```

Release Information

Command introduced before Junos OS Release 7.4.

Show route table evpn statement introduced in Junos OS Release 15.1X53-D30 for QFX Series switches.

RELATED DOCUMENTATION

show route summary

show route terse

IN THIS SECTION

Syntax

```
show route terse
<logical-system (all | logical-system-name)>
```

Syntax (EX Series Switches)

```
show route terse
```

Description

Display a high-level summary of the routes in the routing table.

> **NOTE**: For BGP routes, the `show route terse` command displays the local preference attribute and MED instead of the metric1 and metric2 values. This is mostly due to historical reasons.
>
> To display the metric1 and metric2 value of a BGP route, use the `show route extensive` command.

Options

none	Display a high-level summary of the routes in the routing table.
logical-system (all | *logical-system-name*)	(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege Level

view

Output Fields

Table 49 on page 805 describes the output fields for the `show route terse` command. Output fields are listed in the approximate order in which they appear.

Table 49: show route terse Output Fields

Field Name	Field Description
routing-table-name	Name of the routing table (for example, inet.0).
number destinations	Number of destinations for which there are routes in the routing table.
number routes	Number of routes in the routing table and total number of routes in the following states: • `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)
route key	Key for the state of the route: • `+`—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.
`A`	Active route. An asterisk (*) indicates this is the active route.

Table 49: show route terse Output Fields *(Continued)*

Field Name	Field Description
`V`	Validation status of the route: • `?`—Not evaluated. Indicates that the route was not learned through BGP. • `I`—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. • `N`—Unknown. Indicates that the prefix is not among the prefixes or prefix ranges in the database. • `V`—Valid. Indicates that the prefix and autonomous system pair are found in the database.
Destination	Destination of the route.

Table 49: show route terse Output Fields *(Continued)*

Field Name	Field Description
P	Protocol through which the route was learned: • `A`—Aggregate • `B`—BGP • `C`—CCC • `D`—Direct • `G`—GMPLS • `I`—IS-IS • `L`—L2CKT, L2VPN, LDP, Local • `K`—Kernel • `M`—MPLS, MSDP • `O`—OSPF • `P`—PIM • `R`—RIP, RIPng • `S`—Static • `T`—Tunnel
Prf	Preference value of the route. 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.
Metric 1	First metric value in the route. For routes learned from BGP, this is the MED metric.
Metric 2	Second metric value in the route. For routes learned from BGP, this is the IGP metric.

Table 49: show route terse Output Fields *(Continued)*

Field Name	Field Description
Next hop	Next hop to the destination. An angle bracket (>) indicates that the route is the selected route.
AS path	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: • I—IGP. • E—EGP. • ?—Incomplete; typically, the AS path was aggregated.

Sample Output

show route terse

```
user@host> show route terse
inet.0: 10 destinations, 12 routes (10 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A V Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* ? 172.16.1.1/32         O  10         1            >10.0.0.2
  ?                    B 170        100                             I
  unverified                                     >10.0.0.2
* ? 172.16.1.1/32         D   0                      >lo0.2
* V 2.2.0.2/32         B 170        110                             200 I
  valid                                          >10.0.0.2
* ? 10.0.0.0/30        D   0                     >lt-1/2/0.1
  ?                    B 170        100                             I
  unverified                                     >10.0.0.2
* ? 10.0.0.1/32        L   0                      Local
* ? 10.0.0.4/30        B 170        100                             I
  unverified                                     >10.0.0.2
* ? 10.0.0.8/30        B 170        100                             I
```

```
  unverified                                       >10.0.0.2
* I 172.16.1.1/32      B 170         90                              200 I
  invalid                                          >10.0.0.2
* N 192.168.2.3/32     B 170        100                              200 I
  unknown                                          >10.0.0.2
* ? 172.16.233.5/32       O  10          1             MultiRecv
```

Release Information

Command introduced before Junos OS Release 7.4.

test policy

IN THIS SECTION

Syntax

```
test policy policy-name prefix
```

Description

Test a policy configuration to determine which prefixes match routes in the routing table.

NOTE: If you are using the `test policy` command on a logical system, you must first set the CLI to the logical system context. For example, if you want to test a routing policy that is configured on logical system R2, first run the `set cli logical-system R2` command.

Options

policy-name	Name of a policy.
prefix	Destination prefix to match.

Additional Information

All prefixes in the default unicast routing table (inet.0) that match prefixes that are the same as or longer than the specific prefix are processed by the `from` clause in the specified policy. All prefixes accepted by the policy are displayed. The `test policy` command evaluates a policy differently from the BGP import process. When testing a policy that contains an `interface` match condition in the `from` clause, the `test policy` command uses the match condition. In contrast, BGP does not use the `interface` match condition when evaluating the policy against routes learned from internal BGP (IBGP) or external BGP (EGBP) multihop peers.

When testing a policy, you can see the length of time (in microseconds) required to evaluate the policy and the number of times it has been executed by running the `show policy` *`policy-name`* `statistics` command.

Required Privilege Level

view

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

test policy

```
user@host> test policy test-statics 172.16.0.1/8
inet.0: 44 destinations, 44 routes (44 active, 0 holddown, 0 hidden)
Prefixes passing policy:

172.16.3.0/8          *[BGP/170] 16:22:46, localpref 100, from 10.255.255.41
                   AS Path: 50888 I
                    > to 10.11.4.32 via en0.2, label-switched-path l2
172.16.3.1/32         *[IS-IS/18] 2d 00:21:46, metric 0, tag 2
                    > to 10.0.4.7 via fxp0.0
172.16.3.2/32         *[IS-IS/18] 2d 00:21:46, metric 0, tag 2
                    > to 10.0.4.7 via fxp0.0
172.16.3.3/32         *[IS-IS/18] 2d 00:21:46, metric 0, tag 2
                    > to 10.0.4.7 via fxp0.0
172.16.3.4/32         *[IS-IS/18] 2d 00:21:46, metric 0, tag 2
                    > to 10.0.4.7 via fxp0.0
Policy test-statics: 5 prefixes accepted, 0 prefixes rejected
```

Release Information

Command introduced before Junos OS Release 7.4.

RELATED DOCUMENTATION

Understanding Routing Policy Tests

show policy

show route

show route detail

show route extensive

show route terse