Junos® OS for EX Series Ethernet Switches

Routing Options for EX9200 Switches

Release
13.3

Published: 2014-06-11

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

Junos® OS for EX Series Ethernet Switches Routing Options for EX9200 Switches
Release 13.3

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

List of Figures

List of Tables

About the Documentation

Documentation and Release Notes

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

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

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

Supported Platforms

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

- EX Series

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

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

Documentation Conventions

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

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

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

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the **configure** command: user@host> **configure**

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

Convention	Description	Examples
`Fixed-width text like this`	Represents output that appears on the terminal screen.	`user@host>` **show chassis alarms** `No alarms currently active`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User Guide* • RFC 1997, *BGP Communities Attribute*
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• To configure a stub area, include the **stub** statement at the **[edit protocols ospf area area-id]** hierarchy level. • The console port is labeled **CONSOLE**.
< > (angle brackets)	Encloses optional keywords or variables.	**stub <default-metric** ***metric*****>;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** **(*string1* | *string2* | *string3*)**
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	**rsvp { # Required for dynamic MPLS only**
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	**community name members [** ***community-ids*** **]**

Indention and braces ({ })	Identifies a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		
Bold text like this	Represents graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.

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

Convention	Description	Examples
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page at the Juniper Networks Technical Documentation site at http://www.juniper.net/techpubs/index.html, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at https://www.juniper.net/cgi-bin/docbugreport/.
- E-mail—Send your comments to **techpubs-comments@juniper.net**. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

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

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

Self-Help Online Tools and Resources

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

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/

- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: http://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: http://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: http://www.juniper.net/cm/

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

Opening a Case with JTAC

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

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

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

PART 1

Overview

CHAPTER 1

Routing Properties Overview

Protocol-Independent Routing Properties Overview

In Junos OS, routing capabilities and features that are not specific to any particular routing protocol are collectively called protocol-independent routing properties. These features often interact with routing protocols. In many cases, you combine protocol-independent properties and routing policy to achieve a goal. For example, you define a static route using protocol-independent properties, and then, using a routing policy, you can redistribute the static route into a routing protocol, such as BGP, OSPF, or IS-IS.

Protocol-independent routing properties include:

- Static, aggregate, and generated routes
- Bidirectional Forwarding Detection on static routes
- Global preference
- Martian routes
- Routing tables and routing information base (RIB) groups

Related Documentation

- *Examples: Configuring Static Routes*
- *Examples: Creating a Routing Table and Populating It with Routes*

PART 2

Configuration

CHAPTER 2

Configuration Tasks

Examples: Configuring BFD for Static Routes

Understanding BFD for Static Routes

The Bidirectional Forwarding Detection (BFD) protocol is a simple hello mechanism that detects failures in a network. BFD works with a wide variety of network environments and topologies. A pair of routing devices exchanges BFD packets. 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. The BFD failure detection timers have shorter time limits than the static route failure detection mechanisms, so they provide faster detection.

The BFD failure detection timers are adaptive and can be adjusted to be faster or slower. The lower the BFD failure detection timer value, the faster the failure detection and vice versa. For example, the timers can adapt to a higher value if the adjacency fails (that is, the timer detects failures more slowly). Or a neighbor can negotiate a higher value for a timer than the configured value. The timers adapt to a higher value when a BFD session flap occurs more than three times in a span of 15 seconds. A back-off algorithm increases the receive (Rx) interval by two if the local BFD instance is the reason for the session flap. The transmission (Tx) interval is increased by two if the remote BFD instance is the reason for the session flap. You can use the **clear bfd adaptation** command to return BFD interval timers to their configured values. The **clear bfd adaptation** command is hitless, meaning that the command does not affect traffic flow on the routing device.

By default, BFD is supported on single-hop static routes. In Junos OS Release 8.2 and later, BFD also supports multihop static routes.

To enable failure detection, include the bfd-liveness-detection statement in the static route configuration.

In Junos OS Release 9.1 and later, the BFD protocol is supported for IPv6 static routes. Global unicast and link-local IPv6 addresses are supported for static routes. The BFD protocol is not supported on multicast or anycast IPv6 addresses. For IPv6, the BFD protocol supports only static routes and only in Junos OS Release 9.3 and later. IPv6 for BFD is not supported for any other protocol.

To configure the BFD protocol for IPv6 static routes, include the **bfd-liveness-detection** statement at the **[edit routing-options rib inet6.0 static route *destination-prefix*]** hierarchy level.

In Junos OS Release 8.5 and later, you can configure a hold-down interval to specify how long the BFD session must remain up before a state change notification is sent.

To specify the hold-down interval, include the **holddown-interval** statement in the BFD configuration.

You can configure a number in the range from 0 through 255,000 milliseconds. The default is 0. If the BFD session goes down and then comes back up during the hold-down interval, the timer is restarted.

NOTE: If a single BFD session includes multiple static routes, the hold-down interval with the highest value is used.

To specify the minimum transmit and receive intervals for failure detection, include the **minimum-interval** statement in the BFD configuration.

This value represents both the minimum interval after which the local routing device transmits hello packets and the minimum interval after which the routing device expects to receive a reply from the neighbor with which it has established a BFD session. You can configure a number in the range from 1 through 255,000 milliseconds. Optionally, instead of using this statement, you can configure the minimum transmit and receive intervals separately using the **transmit-interval minimum-interval** and **minimum-receive-interval** statements.

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 NSR configured, the minimum interval recommendations are unchanged and depend only on your network deployment.

NOTE: SRX Series devices do not support distributed BFD.

To specify the minimum receive interval for failure detection, include the **minimum-receive-interval** statement in the BFD configuration. This value represents the minimum interval after which the routing device expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a number in the range from 1 through 255,000 milliseconds. Optionally, instead of using this statement, you can configure the minimum receive interval using the **minimum-interval** statement at the **[edit routing-options static route *destination-prefix* bfd-liveness-detection]** hierarchy level.

To specify the number of hello packets not received by the neighbor that causes the originating interface to be declared down, include the **multiplier** statement in the BFD configuration.

The default value is 3. You can configure a number in the range from 1 through 255.

To specify a threshold for detecting the adaptation of the detection time, include the **threshold** statement in the BFD configuration.

When the BFD session detection time adapts to a value equal to or higher than the threshold, a single trap and a system log message are sent. The detection time is based on the multiplier of the **minimum-interval** or the **minimum-receive-interval** value. The threshold must be a higher value than the multiplier for either of these configured values. For example if the **minimum-receive-interval** is 300 ms and the **multiplier** is 3, the total detection time is 900 ms. Therefore, the detection time threshold must have a value higher than 900.

To specify the minimum transmit interval for failure detection, include the **transmit-interval minimum-interval** statement in the BFD configuration.

This value represents the minimum interval after 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. Optionally, instead of using this statement, you can configure the minimum transmit interval using the **minimum-interval** statement at the **[edit routing-options static route *destination-prefix* bfd-liveness-detection]** hierarchy level.

To specify the threshold for the adaptation of the transmit interval, include the **transmit-interval threshold** statement in the BFD configuration.

The threshold value must be greater than the transmit interval. When the BFD session transmit time adapts to a value greater than the threshold, a single trap and a system log message are sent. The detection time is based on the multiplier of the value for the **minimum-interval** or the **minimum-receive-interval** statement at the **[edit routing-options static route *destination-prefix* bfd-liveness-detection]** hierarchy level. The threshold must be a higher value than the multiplier for either of these configured values.

To specify the BFD version, include the **version** statement in the BFD configuration. The default is to have the version detected automatically.

To include an IP address for the next hop of the BFD session, include the **neighbor** statement in the BFD configuration.

NOTE: You must configure the neighbor statement if the next hop specified is an interface name. If you specify an IP address as the next hop, that address is used as the neighbor address for the BFD session.

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 in the BFD configuration.

NOTE: We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation in your network.

NOTE: If BFD is configured only on one end of a static route, the route is removed from the routing table. BFD establishes a session when BFD is configured on both ends of the static route.

BFD is not supported on ISO address families in static routes. BFD does support IS-IS.

If you configure graceful Routing Engine switchover (GRES) at the same time as BFD, GRES does not preserve the BFD state information during a failover.

Junos OS also supports BFD over multihop static routes. For example, you can configure BFD over a Layer 3 path to provide path integrity over that path. You can limit the number of hops by specifying the time to live (TTL).

To configure BFD over multihop static routes, include the following statements:

```
static route destination-prefix {
  bfd-liveness-detection {
    local-address ip-address;
    minimum-receive-ttl number;
  }
}
```

To specify the source address for the multihop static route and to enable multihop BFD support, include the **local-address** statement.

To specify the number of hops, include the **minimum-receive-ttl** statement. You must configure this statement for a multihop BFD session. You can configure a value in the range from 1 through 255. It is optional for a single-hop BFD session. If you configure the **minimum-receive-ttl** statement for a single-hop session, the value must be 255.

On M Series and T Series platforms only, starting in Junos OS Release 12.3, multihop BFD runs on the CPU in the FPC, DPC, or MPC. This is referred to as *distributed BFD*. Previously, multihop BFD ran from the Routing Engine.

Example: Configuring BFD for Static Routes

This example shows how to configure Bidirectional Forwarding Detection (BFD) for static routes.

Requirements

In this example, no special configuration beyond device initialization is required.

Overview

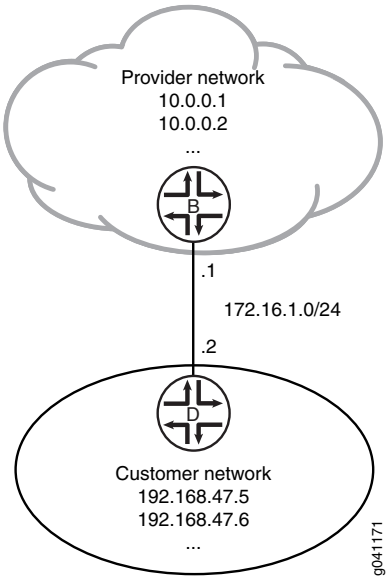
There are many practical applications for static routes. Static routing is often used at the network edge to support attachment to stub networks, which, given their single point of entry and egress, are well suited to the simplicity of a static route. In Junos OS, static routes have a global preference of 5. Static routes are activated if the specified next hop is reachable.

In this example, you configure the static route 192.168.47.0/24 from the provider network to the customer network, using the next-hop address of 172.16.1.2. You also configure a static default route of 0.0.0.0/0 from the customer network to the provider network, using a next-hop address of 172.16.1.1.

For demonstration purposes, some loopback interfaces are configured on Device B and Device D. These loopback interfaces provide addresses to ping and thus verify that the static routes are working.

Figure 1 on page 12 shows the sample network.

Figure 1: Customer Routes Connected to a Service Provider



Configuration

CLI Quick Configuration

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

Device B

```
set interfaces ge-1/2/0 unit 0 description B->D
set interfaces ge-1/2/0 unit 0 family inet address 172.16.1.1/24
set interfaces lo0 unit 57 family inet address 10.0.0.1/32
set interfaces lo0 unit 57 family inet address 10.0.0.2/32
set routing-options static route 192.168.47.0/24 next-hop 172.16.1.2
set routing-options static route 192.168.47.0/24 bfd-liveness-detection minimum-interval
    1000
set protocols bfd traceoptions file bfd-trace
set protocols bfd traceoptions flag all
```

Device D

```
set interfaces ge-1/2/0 unit 1 description D->B
set interfaces ge-1/2/0 unit 1 family inet address 172.16.1.2/24
set interfaces lo0 unit 2 family inet address 192.168.47.5/32
set interfaces lo0 unit 2 family inet address 192.168.47.6/32
set routing-options static route 0.0.0.0/0 next-hop 172.16.1.1
set routing-options static route 0.0.0.0/0 bfd-liveness-detection minimum-interval 1000
set protocols bfd traceoptions file bfd-trace
set protocols bfd traceoptions flag all
```

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 BFD for static routes:

1. On Device B, configure the interfaces.

```
[edit interfaces]
user@B# set ge-1/2/0 unit 0 description B->D
user@B# set ge-1/2/0 unit 0 family inet address 172.16.1.1/24
user@B# set lo0 unit 57 family inet address 10.0.0.1/32
user@B# set lo0 unit 57 family inet address 10.0.0.2/32
```

2. On Device B, create a static route and set the next-hop address.

```
[edit routing-options]
user@B# set static route 192.168.47.0/24 next-hop 172.16.1.2
```

3. On Device B, configure BFD for the static route.

```
[edit routing-options]
user@B# set static route 192.168.47.0/24 bfd-liveness-detection minimum-interval
  1000
```

4. On Device B, configure tracing operations for BFD.

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

5. If you are done configuring Device B, commit the configuration.

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

6. On Device D, configure the interfaces.

```
[edit interfaces]
user@D# set ge-1/2/0 unit 1 description D->B
user@D# set ge-1/2/0 unit 1 family inet address 172.16.1.2/24
user@D# set lo0 unit 2 family inet address 192.168.47.5/32
user@D# set lo0 unit 2 family inet address 192.168.47.6/32
```

7. On Device D, create a static route and set the next-hop address.

```
[edit routing-options]
user@D# set static route 0.0.0.0/0 next-hop 172.16.1.1
```

8. On Device D, configure BFD for the static route.

```
[edit routing-options]
user@D# set static route 0.0.0.0/0 bfd-liveness-detection minimum-interval 1000
```

9. On Device D, configure tracing operations for BFD.

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

10. If you are done configuring Device D, commit the configuration.

```
[edit]
```

```
user@D# commit
```

Results

Confirm your configuration by issuing the **show interfaces**, **show protocols**, and **show routing-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

Device B

```
user@B# show interfaces
ge-1/2/0 {
    unit 0 {
        description B->D;
        family inet {
            address 172.16.1.1/24;
        }
    }
}
lo0 {
    unit 57 {
        family inet {
            address 10.0.0.1/32;
            address 10.0.0.2/32;
        }
    }
}
```

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

```
user@B# show routing-options
static {
    route 192.168.47.0/24 {
        next-hop 172.16.1.2;
        bfd-liveness-detection {
            minimum-interval 1000;
        }
    }
}
```

Device D

```
user@D# show interfaces
ge-1/2/0 {
    unit 1 {
        description D->B;
        family inet {
            address 172.16.1.2/24;
        }
    }
}
lo0 {
    unit 2 {
        family inet {
```

```
        address 192.168.47.5/32;
        address 192.168.47.6/32;
      }
    }
  }

  user@D# show routing-options
  static {
    route 0.0.0.0/0 {
      next-hop 172.16.1.1;
      bfd-liveness-detection {
        minimum-interval 1000;
      }
    }
  }
```

Verification

Confirm that the configuration is working properly.

- Verifying That BFD Sessions Are Up on page 15
- Viewing Detailed BFD Events on page 16

Verifying That BFD Sessions Are Up

Purpose Verify that the BFD sessions are up, and view details about the BFD sessions.

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

```
user@B> show bfd session extensive
                                                  Detect   Transmit
Address                 State     Interface       Time     Interval  Multiplier
172.16.1.2              Up        lt-1/2/0.0      3.000     1.000        3
 Client Static, TX interval 1.000, RX interval 1.000
 Session up time 00:14:30
 Local diagnostic None, remote diagnostic None
 Remote state Up, version 1
 Replicated, routing table index 172
 Min async interval 1.000, min slow interval 1.000
 Adaptive async TX interval 1.000, RX interval 1.000
 Local min TX interval 1.000, minimum RX interval 1.000, multiplier 3
 Remote min TX interval 1.000, min RX interval 1.000, multiplier 3
 Local discriminator 2, remote discriminator 1
 Echo mode disabled/inactive

1 sessions, 1 clients
Cumulative transmit rate 1.0 pps, cumulative receive rate 1.0 pps

user@D> show bfd session extensive
                                                  Detect   Transmit
Address                 State     Interface       Time     Interval  Multiplier
172.16.1.1              Up        lt-1/2/0.1      3.000     1.000        3
 Client Static, TX interval 1.000, RX interval 1.000
 Session up time 00:14:35
 Local diagnostic None, remote diagnostic None
 Remote state Up, version 1
 Replicated, routing table index 170
 Min async interval 1.000, min slow interval 1.000
```

```
 Adaptive async TX interval 1.000, RX interval 1.000
 Local min TX interval 1.000, minimum RX interval 1.000, multiplier 3
 Remote min TX interval 1.000, min RX interval 1.000, multiplier 3
 Local discriminator 1, remote discriminator 2
 Echo mode disabled/inactive

1 sessions, 1 clients
Cumulative transmit rate 1.0 pps, cumulative receive rate 1.0 pps
```

Meaning The **TX interval 1.000, RX interval 1.000** output represents the setting configured with the **minimum-interval** statement. All of the other output represents the default settings for BFD. To modify the default settings, include the optional statements under the **bfd-liveness-detection** statement.

Viewing Detailed BFD Events

Purpose View the contents of the BFD trace file to assist in troubleshooting, if needed.

Action From operational mode, enter the **file show /var/log/bfd-trace** command.

```
user@B> file show /var/log/bfd-trace
Nov 23 14:26:55    Data (9) len 35: (hex) 42 46 44 20 70 65 72 69 6f 64 69 63 20
 78 6d 69 74 20 72
Nov 23 14:26:55 PPM Trace: BFD periodic xmit rt tbl index 172
Nov 23 14:26:55 Received Downstream TraceMsg (22) len 108:
Nov 23 14:26:55    IfIndex (3) len 4: 0
Nov 23 14:26:55    Protocol (1) len 1: BFD
Nov 23 14:26:55    Data (9) len 83: (hex) 70 70 6d 64 5f 62 66 64 5f 73 65 6e 64
 6d 73 67 20 3a 20
Nov 23 14:26:55 PPM Trace: ppmd_bfd_sendmsg : socket 12 len 24, ifl 78 src
172.16.1.1 dst 172.16.1.2 errno 65
Nov 23 14:26:55 Received Downstream TraceMsg (22) len 93:
Nov 23 14:26:55    IfIndex (3) len 4: 0
Nov 23 14:26:55    Protocol (1) len 1: BFD
Nov 23 14:26:55    Data (9) len 68: (hex) 42 46 44 20 70 65 72 69 6f 64 69 63 20
 78 6d 69 74 20 74
```

Meaning BFD messages are being written to the trace file.

Example: Enabling BFD on Qualified Next Hops in Static Routes

This example shows how to configure a static route with multiple possible next hops. Each next hop has Bidirectional Forwarding Detection (BFD) enabled.

Requirements

In this example, no special configuration beyond device initialization is required.

Overview

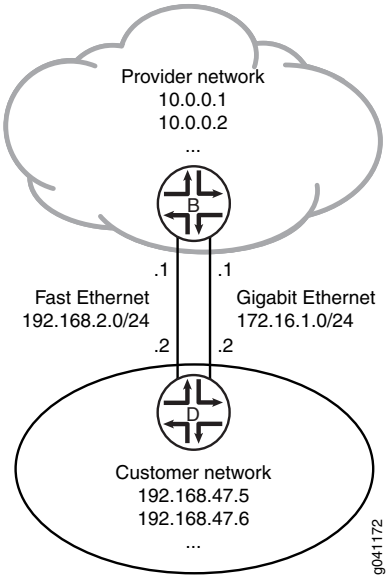
In this example, Device B has the static route **192.168.47.0/24** with two possible next hops. The two next hops are defined using two **qualified-next-hop** statements. Each next hop has BFD enabled.

BFD is also enabled on Device D because BFD must be enabled on both ends of the connection.

A next hop is included in the routing table if the BFD session is up. The next hop is removed from the routing table if the BFD session is down.

See Figure 2 on page 17.

Figure 2: BFD Enabled on Qualified Next Hops



Configuration

CLI Quick Configuration

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

Device B

```
set interfaces fe-0/1/0 unit 2 description secondary-B->D
set interfaces fe-0/1/0 unit 2 family inet address 192.168.2.1/24
set interfaces ge-1/2/0 unit 0 description B->D
set interfaces ge-1/2/0 unit 0 family inet address 172.16.1.1/24
set routing-options static route 192.168.47.0/24 qualified-next-hop 192.168.2.2
    bfd-liveness-detection minimum-interval 60
set routing-options static route 192.168.47.0/24 qualified-next-hop 172.16.1.2
    bfd-liveness-detection minimum-interval 60
```

Device D

```
set interfaces fe-0/1/0 unit 3 description secondary-D->B
set interfaces fe-0/1/0 unit 3 family inet address 192.168.2.2/24
```

```
set interfaces ge-1/2/0 unit 1 description D->B
set interfaces ge-1/2/0 unit 1 family inet address 172.16.1.2/24
set routing-options static route 0.0.0.0/0 qualified-next-hop 192.168.2.1
set routing-options static route 0.0.0.0/0 qualified-next-hop 172.16.1.1
set routing-options static route 0.0.0.0/0 bfd-liveness-detection minimum-interval 60
```

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a static route with two possible next hops, both with BFD enabled:

1. On Device B, configure the interfaces.

```
[edit interfaces fe-0/1/0]
user@B# set unit 2 description secondary-B->D
user@B# set unit 2 family inet address 192.168.2.1/24

[edit interfaces ge-1/2/0]
user@B# set unit 0 description B->D
user@B# set unit 0 family inet address 172.16.1.1/24
```

2. On Device B, configure the static route with two next hops, both with BFD enabled.

```
[edit routing-options static route 192.168.47.0/24]
user@B# set qualified-next-hop 192.168.2.2 bfd-liveness-detection minimum-interval
  60
user@B# set qualified-next-hop 172.16.1.2 bfd-liveness-detection minimum-interval
  60
```

3. On Device D, configure the interfaces.

```
[edit interfaces fe-0/1/0]
user@D# set unit 3 description secondary-D->B
user@D# set unit 3 family inet address 192.168.2.2/24

[edit interfaces ge-1/2/0]
user@D# set unit 1 description D->B
user@D# set unit 1 family inet address 172.16.1.2/24
```

4. On Device D, configure a BFD-enabled default static route with two next hops to the provider network.

   In this case, BFD is enabled on the route, not on the next hops.

```
[edit routing-options static route 0.0.0.0/0]
user@D# set qualified-next-hop 192.168.2.1
user@D# set qualified-next-hop 172.16.1.1
user@D# set bfd-liveness-detection minimum-interval 60
```

Results

Confirm your configuration by issuing the **show interfaces** 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@B# show interfaces
fe-0/1/0 {
  unit 2 {
```

```
      description secondary-B->D;
      family inet {
        address 192.168.2.1/24;
      }
    }
  }
  ge-1/2/0 {
    unit 0 {
      description B->D;
      family inet {
        address 172.16.1.1/24;
      }
    }
  }
```

```
user@B# show routing-options
static {
  route 192.168.47.0/24 {
    qualified-next-hop 192.168.2.2 {
      bfd-liveness-detection {
        minimum-interval 60;
      }
    }
    qualified-next-hop 172.16.1.2 {
      bfd-liveness-detection {
        minimum-interval 60;
      }
    }
  }
}
```

```
user@D# show interfaces
fe-0/1/0 {
  unit 3 {
    description secondary-D->B;
    family inet {
      address 192.168.2.2/24;
    }
  }
}
ge-1/2/0 {
  unit 1 {
    description D->B;
    family inet {
      address 172.16.1.2/24;
    }
  }
}
```

```
user@D# show routing-options
static {
  route 0.0.0.0/0 {
    qualified-next-hop 192.168.2.1;
    qualified-next-hop 172.16.1.1;
    bfd-liveness-detection {
      minimum-interval 60;
    }
```

```
    }
  }
```

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

Verification

Confirm that the configuration is working properly.

Checking the Routing Tables

Purpose Make sure that the static route appears in the routing table on Device B with two possible next hops.

Action

```
user@B> show route 192.168.47.0 extensive
inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
192.168.47.0/24 (1 entry, 1 announced)
TSI:
KRT in-kernel 192.168.47.0/24 -> {192.168.2.2}
        *Static Preference: 5
                Next hop type: Router
                Address: 0x9334010
                Next-hop reference count: 1
                Next hop: 172.16.1.2 via ge-1/2/0.0
                Next hop: 192.168.2.2 via fe-0/1/0.2, selected
                State: <Active Int Ext>
                Age: 9
                Task: RT
                Announcement bits (1): 3-KRT
                AS path: I
```

Meaning Both next hops are listed. The next hop 192.168.2.2 is the selected route.

Verifying the BFD Sessions

Purpose Make sure that the BFD sessions are up.

Action

```
user@B> show bfd session

                                           Detect   Transmit
Address          State     Interface       Time     Interval  Multiplier
172.16.1.2       Up        ge-1/2/0.0      0.720     0.240        3
192.168.2.2      Up        fe-0/1/0.2      0.720     0.240        3

2 sessions, 2 clients
Cumulative transmit rate 8.3 pps, cumulative receive rate 8.3 pps
```

Meaning The output shows that the BFD sessions are up.

Removing BFD from Device D

Purpose Demonstrate what happens when the BFD session is down for both next hops.

Action

1. Deactivate BFD on Device D.

```
[edit routing-options static route 0.0.0.0/0]
user@D# deactivate bfd-liveness-detection
user@D# commit
```

2. Rerun the **show bfd session** command on Device B.

```
user@B> show bfd session

                                                  Detect   Transmit
Address                 State     Interface      Time     Interval  Multiplier
172.16.1.2               Down      ge-1/2/0.0     3.000     1.000        3
192.168.2.2              Down      fe-0/1/0.2     3.000     1.000        3

2 sessions, 2 clients
Cumulative transmit rate 2.0 pps, cumulative receive rate 2.0 pps
```

3. Rerun the **show route 192.168.47.0** command on Device B.

```
user@B> show route 192.168.47.0
```

Meaning As expected, when the BFD sessions are down, the static route is removed from the routing table.

Removing BFD from One Next Hop

Purpose Demonstrate what happens when only one next hop has BFD enabled.

Action

1. If it is not already deactivated, deactivate BFD on Device D.

```
[edit routing-options static route 0.0.0.0/0]
user@D# deactivate bfd-liveness-detection
user@D# commit
```

2. Deactivate BFD on one of the next hops on Device B.

```
[edit routing-options static route 192.168.47.0/24 qualified-next-hop 172.16.1.2]
user@B# deactivate bfd-liveness-detection
user@B# commit
```

3. Rerun the **show bfd session** command on Device B.

```
user@B> show bfd session

                                       Detect   Transmit
Address      State     Interface       Time     Interval  Multiplier
192.168.2.2   Down      fe-0/1/0.2      3.000     1.000        3
```

4. Rerun the **show route 192.168.47.0 extensive** command on Device B.

```
user@B> show route 192.168.47.0 extensive

inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
192.168.47.0/24 (1 entry, 1 announced)
TSI:
```

```
KRT in-kernel 192.168.47.0/24 -> {172.16.1.2}
        *Static Preference: 5
                Next hop type: Router, Next hop index: 624
                Address: 0x92f0178
                Next-hop reference count: 3
                Next hop: 172.16.1.2 via ge-1/2/0.0, selected
                State: <Active Int Ext>
                Age: 2:36
                Task: RT
                Announcement bits (1): 3-KRT
                AS path: I
```

Meaning As expected, the BFD session is down for the 192.168.2.2 next hop. The 172.16.1.2 next hop remains in the routing table, and the route remains active, because BFD is not a condition for this next hop to remain valid.

Related Documentation

- Example: Configuring BFD Authentication for Static Routes on page 22
- *Example: Configuring BFD for OSPF*
- *Example: Configuring BFD for BGP*
- *Example: Configuring BFD for IS-IS*
- *Configuring PIM and the Bidirectional Forwarding Detection (BFD) Protocol*

Example: Configuring BFD Authentication for Static Routes

Understanding BFD Authentication for Static Routes

Bidirectional Forwarding Detection (BFD) enables rapid detection of communication failures between adjacent systems. By default, authentication for BFD sessions is disabled. However, when you run BFD over Network Layer protocols, the risk of service attacks can be significant.

NOTE: **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 IPv4 and IPv6 static routes. BFD authentication is not supported on MPLS OAM sessions. BFD authentication is only supported in the Canada and United States version of the Junos OS image and is not available in the export version.

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

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

BFD Authentication Algorithms

Junos OS supports the following algorithms for BFD authentication:

- **simple-password**—Plain-text password. One to 16 bytes of plain text are used to authenticate the BFD session. One or more passwords 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 (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algorithms might go down after a switchover.

Security Authentication Keychains

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

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

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

Strict Versus Loose Authentication

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

Example: Configuring BFD Authentication for Static Routes

This example shows how to configure Bidirectional Forwarding Detection (BFD) authentication for static routes.

Requirements

Junos OS Release 9.6 or later (Canda and United States version).

BFD authentication is only supported in the Canada and United States version of the Junos OS image and is not available in the export version.

Overview

You can configure authentication for BFD sessions running over IPv4 and IPv6 static routes. Routing instances and logical systems are also supported.

The following steps are needed to configure authentication on a BFD session:

1. Specify the BFD authentication algorithm for the static route.
2. Associate the authentication keychain with the static route.

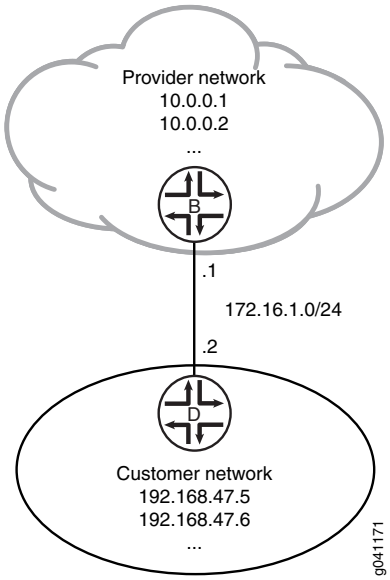
3. Configure the related security authentication keychain. This must be configured on the main router.

TIP: We recommend that you specify loose authentication checking if you are transitioning from nonauthenticated sessions to authenticated sessions.

```
[edit]
user@host> set routing-options static route ipv4 bfd-liveness-detection
  authentication loose-check
```

Figure 3 on page 25 shows the sample network.

Figure 3: Customer Routes Connected to a Service Provider



Configuration

CLI Quick Configuration

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

Device B

```
set interfaces ge-1/2/0 unit 0 description B->D
set interfaces ge-1/2/0 unit 0 family inet address 172.16.1.1/24
set interfaces lo0 unit 57 family inet address 10.0.0.1/32
set interfaces lo0 unit 57 family inet address 10.0.0.2/32
set routing-options static route 192.168.47.0/24 next-hop 172.16.1.2
set routing-options static route 192.168.47.0/24 bfd-liveness-detection minimum-interval
  1000
set routing-options static route 192.168.47.0/24 bfd-liveness-detection authentication
  key-chain bfd-kc4
set routing-options static route 192.168.47.0/24 bfd-liveness-detection authentication
  algorithm keyed-sha-1
```

```
set security authentication-key-chains key-chain bfd-kc4 key 5 secret
  "$9$JhZHmn6Ap0In/9ApOcSs24oaZikPfT3wY24ZG.mz36AtOIEyMWxSrlKvM-dbs2a
  DkP5Ft0IQFclev7N"
set security authentication-key-chains key-chain bfd-kc4 key 5 start-time
  "2011-1-1.12:00:00 -0800"
```

Device D

```
set interfaces ge-1/2/0 unit 1 description D->B
set interfaces ge-1/2/0 unit 1 family inet address 172.16.1.2/24
set interfaces lo0 unit 2 family inet address 192.168.47.5/32
set interfaces lo0 unit 2 family inet address 192.168.47.6/32
set routing-options static route 0.0.0.0/0 next-hop 172.16.1.1
set routing-options static route 0.0.0.0/0 bfd-liveness-detection minimum-interval 1000
set routing-options static route 0.0.0.0/0 bfd-liveness-detection authentication key-chain
  bfd-kc4
set routing-options static route 0.0.0.0/0 bfd-liveness-detection authentication algorithm
  keyed-sha-1
set security authentication-key-chains key-chain bfd-kc4 key 5 secret
  "$9$JhZHmn6Ap0In/9ApOcSs24oaZikPfT3wY24ZG.mz36AtOIEyMWxSrlKvM-dbs2a
  DkP5Ft0IQFclev7N"
set security authentication-key-chains key-chain bfd-kc4 key 5 start-time
  "2011-1-1.12:00:00 -0800"
```

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 BFD for static routes:

1. On Device B, configure the interfaces.

   ```
   [edit interfaces]
   user@B# set ge-1/2/0 unit 0 description B->D
   user@B# set ge-1/2/0 unit 0 family inet address 172.16.1.1/24


   user@B# set lo0 unit 57 family inet address 10.0.0.1/32
   user@B# set lo0 unit 57 family inet address 10.0.0.2/32
   ```

2. On Device B, create a static route and set the next-hop address.

   ```
   [edit routing-options]
   user@B# set static route 192.168.47.0/24 next-hop 172.16.1.2
   ```

3. On Device B, configure BFD for the static route.

   ```
   [edit routing-options]
   user@B# set static route 192.168.47.0/24 bfd-liveness-detection minimum-interval
     1000
   ```

4. On Device B, specify the algorithm (**keyed-md5**, **keyed-sha-1**, **meticulous-keyed-md5**, **meticulous-keyed-sha-1**, or **simple-password**) to use for BFD authentication on the static route.

   ```
   [edit routing-options]
   user@B# set static route 192.168.47.0/24 bfd-liveness-detection authentication
     algorithm keyed-sha-1
   ```

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

5. On Device B, specify the keychain to be used to associate BFD sessions on the specified route with the unique security authentication keychain attributes.

   This should match the keychain name configured at the **[edit security authentication key-chains]** hierarchy level.

   ```
   [edit routing-options]
   user@B# set static route 192.168.47.0/24 bfd-liveness-detection authentication
     key-chain bfd-kc4
   ```

6. On Device B, specify the unique security authentication information for BFD sessions:

   - The matching keychain name as specified in Step 5.
   - 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-kc4]
   user@B# set key 5 secret
     "$9$JhZHmn6Ap0In/9ApOcSs24oaZikPfT3wY24ZG.mz36AtOIEyMWxSrlKvM-dbs2a
     DkP5Ft0IQFclev7N"
   user@B# set key 5 start-time "2011-1-1.12:00:00 -0800"
   ```

7. If you are done configuring Device B, commit the configuration.

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

8. Repeat the configuration on Device D.

   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.

Results

Confirm your configuration by issuing the **show interfaces**, **show routing-options**, and **show security** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

Device B

```
user@B# show interfaces
ge-1/2/0 {
  unit 0 {
    description B->D;
    family inet {
```

```
        address 172.16.1.1/24;
      }
    }
  }
  lo0 {
    unit 57 {
      family inet {
        address 10.0.0.1/32;
        address 10.0.0.2/32;
      }
    }
  }

  user@B# show routing-options
  static {
    route 192.168.47.0/24 {
      next-hop 172.16.1.2;
      bfd-liveness-detection {
        minimum-interval 1000;
        authentication {
          key-chain bfd-kc4;
          algorithm keyed-sha-1;
        }
      }
    }
  }

  user@B# show security
  authentication-key-chains {
    key-chain bfd-kc4 {
      key 5 {
        secret
          "$9$JhZHmn6Ap0In/9ApOcSs24oaZikPfT3wY24ZG.mz36AtOIEyMWxSrlKvM-dbs2a
          DkP5Ft0IQFclev7N"; ## SECRET-DATA
        start-time "2011-1-1.12:00:00 -0800";
      }
    }
  }
```

Verification

Confirm that the configuration is working properly.

- Verifying That BFD Sessions Are Up on page 28
- Viewing Details About the BFD Session on page 29
- Viewing Extensive BFD Session Information on page 29

Verifying That BFD Sessions Are Up

Purpose Verify that the BFD sessions are up.

Action From operational mode, enter the **show bfd session** command.

```
user@B> show bfd session
                                                  Detect   Transmit
Address                 State     Interface       Time     Interval  Multiplier
```

```
172.16.1.2              Up        ge-1/2/0.0     3.000     1.000        3

1 sessions, 1 clients
Cumulative transmit rate 1.0 pps, cumulative receive rate 1.0 pps
```

Meaning The command output shows that the BFD session is up.

Viewing Details About the BFD Session

Purpose View details about the BFD sessions and make sure that authentication is configured.

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

```
user@B> show bfd session detail
                                                  Detect   Transmit
Address                 State     Interface      Time     Interval  Multiplier
172.16.1.2              Up        ge-1/2/0.0     3.000     1.000        3
 Client Static, TX interval 1.000, RX interval 1.000, Authenticate
 Session up time 00:53:58
 Local diagnostic NbrSignal, remote diagnostic None
 Remote state Up, version 1
 Logical system 9, routing table index 22

1 sessions, 1 clients
Cumulative transmit rate 1.0 pps, cumulative receive rate 1.0 pps
```

Meaning In the command output, **Authenticate** is displayed to indicate that BFD authentication is configured.

Viewing Extensive BFD Session Information

Purpose View more detailed information about the BFD sessions.

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

```
user@B> show bfd session extensive
Address                 State     Interface      Time     Interval  Multiplier
172.16.1.2              Up        ge-1/2/0.0     3.000     1.000        3
 Client Static, TX interval 1.000, RX interval 1.000, Authenticate
        keychain bfd-kc4, algo keyed-sha-1, mode strict
 Session up time 01:39:45
 Local diagnostic NbrSignal, remote diagnostic None
 Remote state Up, version 1
 Logical system 9, routing table index 22
 Min async interval 1.000, min slow interval 1.000
 Adaptive async TX interval 1.000, RX interval 1.000
 Local min TX interval 1.000, minimum RX interval 1.000, multiplier 3
 Remote min TX interval 1.000, min RX interval 1.000, multiplier 3
 Local discriminator 3, remote discriminator 4
 Echo mode disabled/inactive
 Authentication enabled/active, keychain bfd-kc4, algo keyed-sha-1, mode strict

1 sessions, 1 clients
Cumulative transmit rate 1.0 pps, cumulative receive rate 1.0 pps
```

Meaning In the command output, **Authenticate** is displayed to indicate that BFD authentication is configured. The output for the **extensive** command provides the keychain name, the authentication algorithm, and the mode for each client in the session.

Related Documentation

- Examples: Configuring BFD for Static Routes on page 7

CHAPTER 3

Configuration Statements

[edit routing-options] Hierarchy Level

Several statements in the **[edit routing-options** hierarchy are valid at numerous locations within the hierarchy. To make the complete hierarchy easier to read, the repeated statements are listed in "Common Routing Options" on page 31 and that section is referenced at the appropriate locations in "Complete [edit routing-options] Hierarchy" on page 33.

Common Routing Options

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in "Complete [edit routing-options] Hierarchy" on page 33 instead of the statements being repeated.

- **[edit routing-options aggregate defaults]**
- **[edit routing-options aggregate route** *ip-prefix</prefix-length>***]**
- **[edit routing-options generate defaults]**
- **[edit routing-options generate route** *ip-prefix</prefix-length>***]**
- **[edit routing-options static defaults]**
- **[edit routing-options static route** *ip-prefix</prefix-length>***]**

The common routing options are as follows:

```
(active | passive);
as-path {
   aggregator as-number address;
   atomic-aggregate;
   origin (egp | igp | incomplete);
   path path-identifier;
```

```
}
color metric <type metric-type>;
color2 metric <type metric-type>;
community [ community-id no-advertise no-export no-export-subconfed ];
metric metric <type metric-type>;
metric2 metric <type metric-type>;
metric3 metric <type metric-type>;
metric4 metric <type metric-type>;
passive;
preference preference-value <type metric-type>;
preference2 preference-value <type metric-type>;
tag metric <type metric-type>;
tag2 metric <type metric-type>;
```

Complete [edit routing-options] Hierarchy

The statement hierarchy in this section can also be included at the **[edit logical-systems *logical-system-name*]** hierarchy level.

```
routing-options {
  access {
    route ip-prefix</prefix-length> {
      metric metric;
      next-hop [ addresses ];
      preference preference-value;
      qualified-next-hop address;
      tag route-tag;
    }
  }
  access-internal {
    route ip-prefix</prefix-length> {
      next-hop [ addresses ];
      qualified-next-hop address;
      tag route-tag;
    }
  }
  admin-groups-extended group-name {
    group-value group-identifier;
  }
  admin-groups-extended-range {
    maximum maximum-number;
    mininum minimum-number;
  }
  aggregate {
    defaults {
      ... statements in Common Routing Options on page 31 PLUS ...
      (brief | full);
      discard;
    }
    route ip-prefix</prefix-length> {
      ... statements in Common Routing Options on page 31 PLUS ...
      (brief | full);
      discard;
      policy [ policy-names ];
    }
  }
  auto-export {
    disable;
    family inet {
      disable;
      flow {
        disable;
        rib-group rib-group;
      }
      multicast {
        disable;
        rib-group rib-group;
      }
      unicast {
```

```
        disable;
        rib-group rib-group;
      }
    }
    family inet6 {
      disable;
      multicast {
        disable;
        rib-group rib-group;
      }
      unicast {
        disable;
        rib-group rib-group;
      }
    }
    family iso {
      disable;
      unicast {
        disable;
        rib-group rib-group;
      }
    }
    traceoptions {
      file filename <files number> <size maximum-file-size> <world-readable |
        no-world-readable>;
      flag flag <flag-modifier> <disable>;
    }
}
autonomous-system autonomous-system <asdot-notation> <loops number>;
no-bfd-triggered-local-repair;
bgp-orf-cisco-mode;
bmp {
  memory-limit bytes;
  station-address (ip-address | name);
  station-port-number port-number;
  statistics-timeout seconds;
}
confederation as-number members [ as-numbers ];
dynamic-tunnels tunnel-name {
  destination-networks prefix;
  gre;
  rsvp-te entry-name {
    destination-networks network-prefix;
    label-switched-path-template {
      default-template;
      template-name;
    }
  }
  source-address address;
}
fate-sharing {
  group group-name {
    cost value;
    from {
      address <to address>;
    }
```

```
    }
  }
  flow {
    firewall-install-disable;
    route name {
      match {
        destination address;
        destination-port [ afs bgp biff bootpc bootps cmd cvspserver dhcp domain eklogin
          ekshell exec finger ftp ftp-data http https ident imap kerberos-sec klogin kpasswd
          krb-prop krbupdate kshell ldap ldp login mobileip-agent mobilip-mn msdp
          netbios-dgm netbios-ns netbios-ssn nfsd nntp ntalk ntp pop3 pptp printer radacct
          radius rip rkinit smtp snmp snmptrap snpp socks ssh sunrpc syslog tacacs
          tacacs-ds talk telnet tftp timed who xdmcp ];
        dscp [ code-points ];
        fragment [ don't-fragment first-fragment is-fragment last-fragment
          not-a-fragment ];
        icmp-code [ communication-prohibited-by-filtering destination-host-prohibited
          destination-host-unknow fragmentation-needed host-precedence-violation
          host-unreachable host-unreachable-for-tos ip-header-bad network-unreachable
          network-unreachable-for-tos port-unreachable precedence-cutoff-in-effect
          protocol-unreachable redirect-for-host redirect-for-network
          redirect-for-tos-and-host redirect-for-tos-and-net required-option-missing
          source-host-isolated source-route-failed ttl-eq-zero-during-reassembly
          ttl-eq-zero-during-transit ];
        icmp-type [ echo-reply echo-request info-reply info-request mask-reply
          mask-request parameter-problem redirect router-advertisement router-solicit
          source-quench time-exceeded timestamp timestamp-reply unreachable ];
        packet-length [ values ];
        port [ ... same values as for the preceding destination-port statement ... ];
        protocol [ ah egp esp gre icmp igmp ipip ospf pim rsvp sctp tcp udp ];
        source address;
        source-port [ ... same values as for the preceding destination-port statement ... ];
        tcp-flags [ ack fin push rst syn urgent ];
      }
      then {
        (accept | discard);
        community community-name;
        next-term;
        rate-limit value;
        routing-instance routing-instance-name;
        sample;
      }
    }
    term order (legacy | standard);
    validation {
      traceoptions {
        file filename <files number> <size maximum-file-size> <world-readable |
          no-world-readable>;
        flag flag <flag-modifier> <disable>;
      }
    }
  }
  forwarding-table {
    chained-composite-next-hop {
      ingress {
        l3vpn {
```

```
            extended-space;
          }
        }
      }
    }
    export [ policy-name ];
    (indirect-next-hop | no-indirect-next-hop);
    (indirect-next-hop-change-acknowledgements |
      no-indirect-next-hop-change-acknowledgements;)
    krt-nexthop-ack-timeout interval;
    unicast-reverse-path (active-paths | feasible-paths);
}
generate {
    defaults {
      ... statements in Common Routing Options on page 31 PLUS ...
      (brief | full);
      discard;
    }
    route ip-prefix</prefix-length> {
      ... statements in Common Routing Options on page 31 PLUS ...
      (brief | full);
      discard;
      policy [ policy-names ];
    }
}
graceful-restart {
    disable;
    restart-duration seconds;
}
host-fast-reroute {
    global-arp-prefix-limit number;
    global-supplementary-blackout-timer minutes;
}
instance-export [ policy-names ];
instance-import [ policy-names ];
interface interface-name { # In the routing-instance only
    arp-prefix-limit number;
    link-protection;
    supplementary-blackout-timer minutes;
}
interface-routes {
    family (inet | inet6) {
      export {
        lan;
        point-to-point;
      }
      import [ policy-names ];
    }
    rib-group {
      inet group-name;
      inet6 group-name;
    }
}
martians {
    ip-prefix</prefix-length> (exact | longer | orlonger |
      prefix-length-range /minimum-prefix-length–/maximum-prefix-length |
      through ip-prefix</prefix-length> | upto /prefix-length) <allow>;
```

```
}
maximum-paths path-limit <log-only | threshold value> <log-interval seconds>;
maximum-prefixes prefix-limit <log-only | threshold value> <log-interval seconds>;
med-igp-update-interval minutes;
multicast {
  ... the multicast subhierarchy appears after the main [edit routing-options] hierarchy ...
}
nonstop-routing;
options {
  mark seconds;
  syslog {
    level level;
    upto level;
  }
}
ppm {
  no-delegate-processing;
}
resolution {
  rib routing-table-name {
    import [ policy-names ];
    resolution-ribs [ routing-table-names ];
  }
  tracefilter [ filter-policy-names ];
  traceoptions {
    file filename <files number> <size maximum-file-size> <world-readable |
      no-world-readable>;
    flag flag <flag-modifier> <disable>;
  }
}
rib routing-table-name {
  access {
    ... same statements as at the [edit routing-options access] hierarchy level ...
  }
  access-internal {
    ... same statements as at the [edit routing-options access-internal] hierarchy level ...
  }
  aggregate {
    ... same statements as at the [edit routing-options aggregate] hierarchy level ...
  }
  generate {
    ... same statements as at the [edit routing-options generate] hierarchy level ...
  }
  martians {
    ip-prefix</prefix-length> (exact | longer | orlonger |
      prefix-length-range /minimum-prefix-length–/maximum-prefix-length |
      through ip-prefix</prefix-length> | upto /prefix-length) <allow>;
  }
  maximum-paths path-limit <log-only | threshold value> <log-interval seconds>;
  maximum-prefixes prefix-limit <log-only | threshold value> <log-interval seconds>;
  static {
    ... same statements as at the [edit routing-options static] hierarchy level ...
  }
}
rib-groups {
  group-name {
```

```
      export-rib table-name;
      import-policy [ policy-names ];
      import-rib [ table-names ];
    }
  }
  route-distinguisher-id address;
  route-record;
  router-id address;
  source-routing {
    ip;
    ipv6;
  }
  srlg {
    srlg-name {
      srlg-cost srlg-cost;
      srlg-value srlg-value;
    }
  }
  static {
    ... the static subhierarchy appears after the main [edit routing-options] hierarchy ...
  }
  topologies {
    family (inet | inet6) {
      topology topology-name;
    }
  }
  traceoptions {
    file filename <files number> <size maximum-file-size> <world-readable |
      no-world-readable>;
    flag flag <disable>;
  }
  validation {
    group group-name {
      max-sessions number;
      session address {
        hold-time seconds;
        local-address local-ip-address;
        port port-number;
        preference number;
        record-lifetime seconds;
        refresh-time seconds;
      }
    }
    notification-rib value;
    static {
      record destination {
        maximum-length prefix-length {
          origin-autonomous-system as-number {
            validation-state (invalid | valid);
          }
        }
      }
    }
    traceoptions {
      file filename <files number> <size size> <world-readable | no-world-readable>;
      flag flag;
```

```
    }
  }
}

routing-options {
  multicast {
    asm-override-ssm;
    backup-pe-group group-name {
      backups [ addresses ];
      local-address address;
    }
    flow-map flow-map-name {
      bandwidth <bps> <adaptive>;
      forwarding-cache {
        timeout (never <non-discard-entry-only> | minutes);
      }
      policy [ policy-names ];
      redundant-sources [ addresses ];
    }
    forwarding-cache {
      family (inet | inet6) {
        threshold {
          log-warningvalue;
          suppress value <reuse value>;
        }
      threshold {
        log-warning value;
        suppress value <reuse value>;
      }
      timeout minutes;
    }
    interface interface-name {
      maximum-bandwidth bps;
      no-qos-adjust;
      reverse-oif-mapping {
        no-qos-adjust;
      }
      subscriber-leave-timer seconds;
    }
    pim-to-igmp-proxy {
      upstream-interface [ interface-names ];
    }
    pim-to-mld-proxy {
      upstream-interface [ interface-names ];
    }
    rpf-check-policy [ policy-names ];
    scope scope-name {
      interface [ interface-names ];
      prefix ip-prefix</prefix-length>;
    }
    scope-policy [ policy-names ];
    ssm-groups [ ip-prefix</prefix-length> ];
    ssm-map ssm-map-name {
      policy [ policy-names ];
      source [ addresses ];
    }
```

```
    traceoptions {
      file filename <files number> <size maximum-file-size> <world-readable |
        no-world-readable>;
      flag flag <disable>;
    }
  }
}

routing-options {
  static {
    defaults {
      ... statements in Common Routing Options on page 31 PLUS ...
      (install | no-install);
      (readvertise | no-readvertise);
      (resolve | no-resolve);
      (retain | no-retain);
    }
    rib-group group-name;
    route destination-prefix {
      ... statements in Common Routing Options on page 31 PLUS ...
      backup-pe-group group-name;
      bfd-liveness-detection {
        detection-time {
          threshold milliseconds;
        }
        holddown-interval milliseconds;
        local-address ip-address;
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        minimum-receive-ttl milliseconds;
        multiplier number;
        neighbor address;
        no-adaptation;
        transmit-interval {
          minimum-interval milliseconds;
          threshold milliseconds;
        }
        version (1 | automatic);
      }
      (discard | next-hop [ addresses ] | next-table address | receive | reject);
      (install | no-install);
      lsp-next-hop {
        metric metric;
        preference preference;
      }
      p2mp-lsp-next-hop lsp-name {
        metric metric;
        preference preference;
      }
      (readvertise | no-readvertise);
      (resolve | no-resolve);
      (retain | no-retain);
      static-lsp-next-hop lsp-name {
        metric metric;
        preference preference-value;
      }
```

```
        }
    }
}
```

Related Documentation

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

bfd

Syntax

```
bfd {
  traceoptions {
    file filename <files number> <match regular-expression> <size size> <world-readable |
      no-world-readable>;
    flag flag <flag-modifier> <disable>;
  }
}
```

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]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure trace options for Bidirectional Forwarding Protocol (BFD) traffic.

Default If you do not include this statement, no BFD tracing operations are performed.

Options

disable—(Optional) Disable the BFD tracing operation. You can use this option 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. All files are placed in the **/var/log** directory . We recommend that you place global routing protocol tracing output in the **routing-log** file.

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 also must specify a maximum file size with the **size** option.

Range: 2 through 1000 files

Default: 2 files

flag *flag*—Tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements. These are the BFD protocol tracing options:

- **adjacency**—Trace adjacency messages.
- **all**—Trace all options for BFD.
- **error**—Trace all errors.
- **event**—Trace all events.
- **issu**—Trace in-service software upgrade (ISSU) packet activity.

- **nsr-packet**—Trace non-stop-routing (NSR) packet activity.
- **nsr-synchronization**—Trace NSR synchronization events.
- **packet**—Trace all packets.
- **pipe**—Trace pipe messages.
- **pipe-detail**—Trace pipe messages in detail.
- **ppm-packet**—Trace packet activity by periodic packet management (PPM).
- **state**—Trace state transitions.
- **timer**—Trace timer processing.

match *regular-expression*—(Optional) Regular expression for lines to be logged.

no-world-readable—(Optional) Prevent any user from reading 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 also must 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 and trace—To view this statement in the configuration.
routing-control and trace–control—To add this statement to the configuration.

Related Documentation

- Example: Configuring BFD for Static Routes on page 11

bfd-liveness-detection (Routing Options Static Route)

Syntax

```
bfd-liveness-detection {
  authentication {
    algorithm algorithm-name;
    key-chain key-chain-name;
    loose-check;
  }
  detection-time {
    threshold milliseconds;
  }
  holddown-interval milliseconds;
  local-address ip-address;
  minimum-interval milliseconds;
  minimum-receive-interval milliseconds;
  minimum-receive-ttl number;
  multiplier number;
  neighbor address;
  no-adaptation;
  transmit-interval {
    minimum-interval milliseconds;
    threshold milliseconds;
  }
  version (1 | automatic);
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit logical-systems *logical-system-name* routing-options static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-options static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* static route *destination-prefix*],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit routing-instances *routing-instance-name* routing-options static route *destination-prefix*],
[edit routing-instances *routing-instance-name* routing-options static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit routing-options rib *routing-table-name* static route *destination-prefix*],
[edit routing-options rib *routing-table-name* static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit routing-options static route *destination-prefix*],

[edit routing-options static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)]

Release Information Statement introduced before Junos OS Release 7.4.
detection-time threshold and **transmit-interval threshold** options introduced in Junos OS Release 8.2.
local-address statement introduced in Junos OS Release 8.2.
minimum-receive-ttl statement introduced in Junos OS Release 8.2.
Support for logical routers introduced in Junos OS Release 8.3.
holddown-interval statement introduced in Junos OS Release 8.5.
no-adaptation statement introduced in Junos OS Release 9.0.
Support for IPv6 static routes introduced in Junos OS Release 9.1.
authentication algorithm, **authentication key-chain**, and **authentication loose-check** statements introduced in Junos OS Release 9.6.
Statement introduced in Junos OS Release 12.1 for the QFX Series.
Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Description Configure bidirectional failure detection timers and authentication criteria for static routes.

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 may not be 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.

holddown-interval ***milliseconds***—Configure an interval specifying how long a BFD session must remain up before a state change notification is sent. If the BFD session goes down and then comes back up during the hold-down interval, the timer is restarted.

Range: 0 through 255,000

Default: 0

local-address ***ip-address***—Enable a multihop BFD session and configure the source address for the BFD session.

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 configure the minimum transmit and receive intervals separately using the **transmit-interval minimum-interval** and **minimum-receive-interval** statements.

Range: 1 through 255,000

minimum-receive-interval ***milliseconds***—Configure the minimum interval after which the 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 at the **[edit routing-options static route** ***destination-prefix*** **bfd-liveness-detection]** hierarchy level.

Range: 1 through 255,000

minimum-receive-ttl ***number***—Configure the time to live (TTL) for the multihop BFD session.

Range: 1 through 255

Default: 255

multiplier ***number***—Configure number of hello packets not received by the neighbor that causes the originating interface to be declared down.

Range: 1 through 255

Default: 3

neighbor *address*—Configure a next-hop address for the BFD session for a next hop specified as an interface name.

no-adaptation—Specify for 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

transmit-interval minimum-interval *milliseconds*—Configure the minimum interval at which the routing device transmits hello packets to a neighbor with which it has established a BFD session. Optionally, instead of using this statement, you can configure the minimum transmit interval using the **minimum-interval** statement at the **[edit routing-options static route *destination-prefix* bfd-liveness-detection]** hierarchy level.

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.

Related Documentation

- Example: Configuring BFD for Static Routes on page 11
- Example: Configuring BFD Authentication for Static Routes on page 24

PART 3

Administration

CHAPTER 4

Operational Commands

- clear bfd adaptation
- clear bfd session
- show bfd session

clear bfd adaptation

Syntax clear bfd adaptation
<address *session-address*>
<discriminator *discr-number*>

Release Information Command introduced before Junos OS Release 7.4.

Description Clear adaptation for Bidirectional Forwarding Detection (BFD) sessions. BFD is a simple hello mechanism that detects failures in a network. Configured BFD interval timers can change, adapting to network situations. Use this command to return BFD interval timers to their configured values.

The **clear bfd adaptation** command is hitless, meaning that the command does not affect traffic flow on the routing device.

Options **none**—Clear adaptation for all BFD sessions.

address *session-address*—(Optional) Clear adaptation for all BFD sessions matching the specified address.

discriminator *discr-number*—(Optional) Clear adaptation for the local BFD session matching the specified discriminator.

Additional Information For more information, see the description of the **bfd-liveness-detection** configuration statement in the *Junos Routing Protocols Configuration Guide*.

Required Privilege Level clear

List of Sample Output clear bfd adaptation on page 52

Output Fields When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear bfd adaptation

```
user@host> clear bfd adaptation
```


clear bfd session

List of Syntax
Syntax on page 53
Syntax (EX Series Switch and QFX Series) on page 53

Syntax
clear bfd session
<address *session-address*>
<discriminator *discr-number*>
<logical-system (all | *logical-system-name*)>

Syntax (EX Series Switch and QFX Series)
clear bfd session
<address *session-address*>
<discriminator *discr-number*>

Release Information
Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 12.1 for the QFX Series.

Description
Drop one or more Bidirectional Forwarding Detection (BFD) sessions.

Options
none—Drop all BFD sessions.

address *session-address*—(Optional) Drop all BFD sessions matching the specified address.

discriminator *discr-number*—(Optional) Drop the local BFD session matching the specified discriminator.

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

Required Privilege Level
clear

Related Documentation
- show bfd session on page 54

List of Sample Output
clear bfd session on page 53

Output Fields
When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear bfd session

```
user@host> clear bfd session
```

show bfd session

List of Syntax

Syntax on page 54
Syntax (EX Series Switch and QFX Series) on page 54

Syntax

show bfd session
<brief | detail | extensive | summary>
<address *address*>
<client rsvp-oam (brief | detail | extensive | summary) | vpls-oam (brief | detail | extensive | instance *instance-name* | summary)>
<discriminator *discriminator*>
<logical-system (all | *logical-system-name*)>
<prefix *address*>

Syntax (EX Series Switch and QFX Series)

show bfd session
<brief | detail | extensive | summary>
<address *address*>
<client rsvp-oam (brief | detail | extensive | summary) | vpls-oam (brief | detail | extensive | instance *instance-name* | summary)>
<discriminator *discriminator*>
<prefix *address*>

Release Information

Command introduced before Junos OS Release 7.4.
Options **discriminator** and **address** introduced in Junos OS Release 8.2.
Option **prefix** introduced in Junos OS Release 9.0.
Command introduced in Junos OS Release 12.1 for the QFX Series.
Option **client** introduced in Junos OS Release 12.3R3.

Description

Display information about active Bidirectional Forwarding Detection (BFD) sessions.

Options

none—(Same as **brief**) Display information about active BFD sessions.

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

address ***address***—(Optional) Display information about the BFD session for the specified neighbor address.

client rsvp-oam
(brief | detail | extensive | summary)
| vpls-oam
(brief | detail | extensive | instance *instance-name* | summary)—(Optional) Display information about RSVP-OAM or VPLS-OAM BFD sessions in the specified level of output. For VPLS-OAM, display the specified level of output or display information about all of the BFD sessions for the specified VPLS routing instance.

discriminator ***discriminator***—(Optional) Display information about the BFD session using the specified local discriminator.

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

prefix *address*—(Optional) Display information about all of the BFD sessions for the specified LDP forwarding equivalence class (FEC).

Required Privilege Level view

Related Documentation

- clear bfd session on page 53
- Examples: Configuring BFD for Static Routes on page 7
- *Example: Configuring BFD for OSPF*
- *Example: Configuring BFD for BGP*
- *Configuring PIM and the Bidirectional Forwarding Detection (BFD) Protocol*
- *Example: Configuring BFD for IS-IS*

List of Sample Output

show bfd session on page 58
show bfd session brief on page 59
show bfd session detail on page 59
show bfd session detail (with Authentication) on page 59
show bfd session address extensive on page 59
show bfd session client rsvp-oam on page 60
show bfd session client vpls-oam summary on page 60
show bfd session client vpls-oam instance instance-name on page 60
show bfd session extensive on page 60
show bfd session extensive (with Authentication) on page 61
show bfd session summary on page 61

Output Fields Table 3 on page 55 describes the output fields for the **show bfd session** command. Output fields are listed in the approximate order in which they appear.

Table 3: show bfd session Output Fields

Field Name	Field Description	Level of Output
Address	Address on which the BFD session is active.	**brief detail extensive** none
State	State of the BFD session: **Up**, **Down**, **Init** (initializing), or **Failing**.	**brief detail extensive** none
Interface	Interface on which the BFD session is active.	**brief detail extensive** none
Detect Time	Negotiated time interval, in seconds, used to detect BFD control packets.	**brief detail extensive** none
Transmit Interval	Time interval, in seconds, used by the transmitting system to send BFD control packets.	**brief detail extensive** none
Multiplier	Negotiated multiplier by which the time interval is multiplied to determine the detection time for the transmitting system.	**detail extensive**

Table 3: show bfd session Output Fields *(continued)*

Field Name	Field Description	Level of Output
Session up time	How long a BFD session has been established.	**detail extensive**
Client	Protocol for which the BFD session is active: **ISIS**, **OSPF**, or **Static**.	**detail extensive**
TX interval	Time interval, in seconds, used by the host system to transmit BFD control packets.	**brief detail extensive** none
RX interval	Time interval, in seconds, used by the host system to receive BFD control packets.	**brief detail extensive** none
Authenticate	Indicates that BFD authentication is configured.	**detail extensive**
keychain	Name of the security authentication keychain being used by a specific client. BFD authentication information for a client is provided in a single line and includes the **keychain**, **algo**, and **mode** parameters. Multiple clients can be configured on a BFD session.	**extensive**
algo	BFD authentication algorithm being used for a specific client: **keyed-md5**, **keyed-sha-1**, **meticulous-keyed-md5**, **meticulous-keyed-sha-1**, or **simple-password**. BFD authentication information for a client is provided in a single line and includes the **keychain**, **algo**, and **mode** parameters. Multiple clients can be configured on a BFD session.	**extensive**
mode	Level of BFD authentication enforcement being used by a specific client: **strict** or **loose**. Strict enforcement indicates that authentication is configured at both ends of the session (the default). Loose enforcement indicates that one end of the session might not be authenticated. BFD authentication information for a client is provided in a single line and includes the **keychain**, **algo**, and **mode** parameters. Multiple clients can be configured on a BFD session.	**extensive**
Local diagnostic	Local diagnostic information about failing BFD sessions.	**detail extensive**
Remote diagnostic	Remote diagnostic information about failing BFD sessions.	**detail extensive**
Remote state	Reports whether the remote system's BFD packets have been received and whether the remote system is receiving transmitted control packets.	**detail extensive**
Version	BFD version: **0** or **1**.	**extensive**
Replicated	The **replicated** flag appears when nonstop routing or graceful Routing Engine switchover is configured and the BFD session has been replicated to the backup Routing Engine.	**detail extensive**
Min async interval	Minimum amount of time, in seconds, between asynchronous control packet transmissions across the BFD session.	**extensive**

Table 3: show bfd session Output Fields *(continued)*

Field Name	Field Description	Level of Output
Min slow interval	Minimum amount of time, in seconds, between synchronous control packet transmissions across the BFD session.	**extensive**
Adaptive async TX interval	Transmission interval being used because of adaptation.	**extensive**
RX interval	Minimum required receive interval.	**extensive**
Local min TX interval	Minimum amount of time, in seconds, between control packet transmissions on the local system.	**extensive**
Local min RX interval	Minimum amount of time, in seconds, between control packet detections on the local system.	**extensive**
Remote min TX interval	Minimum amount of time, in seconds, between control packet transmissions on the remote system.	**extensive**
Remote min TX interval	Minimum amount of time, in seconds, between control packet detections on the remote system.	**extensive**
Threshold transmission interval	Threshold for notification if the transmission interval increases.	**extensive**
Threshold for detection time	Threshold for notification if the detection time increases.	**extensive**
Local discriminator	Authentication code used by the local system to identify that BFD session.	**extensive**
Remote discriminator	Authentication code used by the remote system to identify that BFD session.	**extensive**
Echo mode	Information about the state of echo transmissions on the BFD session.	**extensive**
Prefix	LDP FEC address associated with the BFD session.	All levels
Egress, Destination	Displays the LDP FEC destination address. This field is displayed only on a router at the egress of an LDP FEC, where the BFD session has an LDP Operation, Administration, and Maintenance (OAM) client.	All levels
Remote is control-plane independent	The BFD session on the remote peer is running on its Packet Forwarding Engine. In this case, when the remote node undergoes a graceful restart, the local peer can help the remote peer with the graceful restart. The following BFD sessions are not distributed to the Packet Forwarding Engine: tunnel-encapsulated sessions, and sessions over integrated routing and bridging (IRB) interfaces.	**extensive**

Table 3: show bfd session Output Fields *(continued)*

Field Name	Field Description	Level of Output
Authentication	Summary status of BFD authentication: • **status**—**enabled/active** indicates authentication is configured and active. **enabled/inactive** indicates authentication is configured but not active. This only occurs when the remote end of the session does not support authentication and loose checking is configured. • **keychain**—Name of the security authentication keychain associated with the specified BFD session. • **algo**—BFD authentication algorithm being used: **keyed-md5**, **keyed-sha-1**, **meticulous-keyed-md5**, **meticulous-keyed-sha-1**, or **simple-password**. • **mode**—Level of BFD authentication enforcement: **strict** or **loose**. Strict enforcement indicates authentication is configured at both ends of the session (the default). Loose enforcement indicates that one end of the session might not be authenticated. This information is only shown if BFD authentication is configured.	**extensive**
Session ID	The BFD session ID number that represents the protection using MPLS fast reroute (FRR) and loop-free alternate (LFA).	**detail extensive**
sessions	Total number of active BFD sessions.	All levels
clients	Total number of clients that are hosting active BFD sessions.	All levels
Cumulative transmit rate	Total number of BFD control packets transmitted per second on all active sessions.	All levels
Cumulative receive rate	Total number of BFD control packets received per second on all active sessions.	All levels
Multi-hop, min-recv-TTL	Minimum time to live (TTL) accepted if the session is configured for multihop.	**extensive**
route table	Route table used if the session is configured for multihop.	**extensive**
local address	Local address of the source used if the session is configured for multihop. The source IP address for outgoing BFD packets from the egress side of an MPLS BFD session is based on the outgoing interface IP address.	**extensive**

Sample Output

show bfd session

```
user@host> show bfd session
                                                  Transmit
Address                 State     Interface     Detect Time  Interval  Multiplier
10.9.1.33               Up        so-7/1/0.0          0.600     0.200       3
10.9.1.29               Up        ge-4/0/0.0          0.600     0.200       3

2 sessions, 2 clients
Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

show bfd session brief

The output for the **show bfd session brief** command is identical to that for the **show bfd session** command. For sample output, see show bfd session on page 58.

show bfd session detail

```
user@host> show bfd session detail
                                                    Transmit
Address              State     Interface     Detect Time  Interval  Multiplier
10.9.1.33            Up        so-7/1/0.0          0.600     0.200      3
  Client OSPF, TX interval 0.200, RX interval 0.200, multiplier 3
  Session up time 3d 00:34
  Local diagnostic None, remote diagnostic None
  Remote state Up, version 1
  Replicated
10.9.1.29            Up        ge-4/0/0.0          0.600     0.200      3
  Client ISIS L2, TX interval 0.200, RX interval 0.200, multiplier 3
  Session up time 3d 00:29, previous down time 00:00:01
  Local diagnostic NbrSignal, remote diagnostic AdminDown
  Remote state Up, version 1

2 sessions, 2 clients
Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

show bfd session detail (with Authentication)

```
user@host> show bfd session detail
                                                    Transmit
Address              State     Interface     Detect Time  Interval  Multiplier
10.9.1.33            Up        so-7/1/0.0          0.600     0.200      3
  Client OSPF, TX interval 0.200, RX interval 0.200, multiplier 3, Authenticate
  Session up time 3d 00:34
  Local diagnostic None, remote diagnostic None
  Remote state Up, version 1
  Replicated
10.9.1.29            Up        ge-4/0/0.0          0.600     0.200      3
  Client ISIS L2, TX interval 0.200, RX interval 0.200, multiplier 3
  Session up time 3d 00:29, previous down time 00:00:01
  Local diagnostic NbrSignal, remote diagnostic AdminDown
  Remote state Up, version 1

2 sessions, 2 clients
Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

show bfd session address extensive

```
user@host> show bfd session 10.255.245.212 extensive
                                                    Transmit
Address              State     Interface     Detect Time  Interval  Multiplier
10.255.245.212       Up                          1.200     0.400      3
  Client Static, TX interval 0.400, RX interval 0.400, multiplier 3
  Session up time 00:17:03, previous down time 00:00:14
  Local diagnostic CtlExpire, remote diagnostic NbrSignal
  Remote state Up, version 1
  Replicated
  Min async interval 0.400, min slow interval 1.000
  Adaptive async tx interval 0.400, rx interval 0.400
  Local min tx interval 0.400, min rx interval 0.400, multiplier 3
  Remote min tx interval 0.400, min rx interval 0.400, multiplier 3
```

```
  Threshold transmission interval 0.000, Threshold for detection time 0.000
  Local discriminator 6, remote discriminator 16
  Echo mode disabled/inactive
  Multi-hop, min-recv-TTL 255, route-table 0, local-address 10.255.245.205


1 sessions, 1 clients
Cumulative transmit rate 2.5 pps, cumulative receive rate 2.5 pps
```

show bfd session client rsvp-oam

```
user@host> show bfd session client rsvp-oam
                                                 Detect   Transmit
 Address                 State     Interface     Time     Interval  Multiplier

 192.168.0.223           Up                      540.000   180.000      3

 1 Up sessions, 0 Down sessions
 1 sessions, 1 clients
 Cumulative transmit rate 0.0 pps, cumulative receive rate 0.0 pps
```

show bfd session client vpls-oam summary

```
user@host> show bfd session client vpls-oam summary
1 Up sessions, 1 Down sessions
 2 sessions, 2 clients
 Cumulative transmit rate 2.0 pps, cumulative receive rate 1.0 pps
```

show bfd session client vpls-oam instance instance-name

```
user@host> show bfd session client vpls-oam instance vpls
                                                 Detect   Transmit
 Address                 State     Interface     Time     Interval  Multiplier

 127.0.0.1               Up        ae9.0         3.000     1.000        3

 1 Up Sessions, 0 Down Sessions
 1 sessions, 1 clients
 Cumulative transmit rate 1.0 pps, cumulative receive rate 1.0 pps
```

show bfd session extensive

```
user@host> show bfd session extensive
10.31.1.2               Up        ge-2/1/8.0     0.030     0.010        3
 Client OSPF realm ospf-v2 Area 0.0.0.0, TX interval 0.010, RX interval 0.010
 Session up time 00:10:13
 Local diagnostic None, remote diagnostic None
 Remote state Up, version 1
 Replicated
 Min async interval 0.010, min slow interval 1.000
 Adaptive async TX interval 0.010, RX interval 0.010
 Local min TX interval 0.010, minimum RX interval 0.010, multiplier 3
 Remote min TX interval 0.010, min RX interval 0.010, multiplier 3
 Local discriminator 12, remote discriminator 4
 Echo mode disabled/inactive
 Remote is control-plane independent
  Session ID: 0x201
  Micro-BFD Session
                                                 Detect   Transmit
Address                 State     Interface     Time     Interval  Multiplier
10.31.2.2               Up        ge-2/1/4.0     0.030     0.010        3
 Client OSPF realm ospf-v2 Area 0.0.0.0, TX interval 0.010, RX interval 0.010
```

```
 Session up time 00:10:14
 Local diagnostic None, remote diagnostic NbrSignal
 Remote state Up, version 1
 Replicated
 Min async interval 0.010, min slow interval 1.000
 Adaptive async TX interval 0.010, RX interval 0.010
 Local min TX interval 0.010, minimum RX interval 0.010, multiplier 3
 Remote min TX interval 0.010, min RX interval 0.010, multiplier 3
 Local discriminator 13, remote discriminator 5
 Echo mode disabled/inactive
 Remote is control-plane independent
  Session ID: 0x202

2 sessions, 2 clients
Cumulative transmit rate 200.0 pps, cumulative receive rate 200.0 pps
```

show bfd session extensive (with Authentication)

```
user@host> show bfd session extensive
                                                  Detect   Transmit
Address                 State     Interface       Time     Interval  Multiplier
192.168.208.26          Up        so-1/0/0.0      2.400     0.800        10
 Client Static, TX interval 0.600, RX interval 0.600, Authenticate
        keychain bfd, algo keyed-md5, mode loose
 Session up time 00:18:07
 Local diagnostic None, remote diagnostic NbrSignal
 Remote state Up, version 1
 Replicated
 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 10
 Remote min TX interval 0.800, min RX interval 0.800, multiplier 3
 Local discriminator 2, remote discriminator 3
 Echo mode disabled/inactive
 Authentication enabled/active, keychain bfd, algo keyed-md5, mode loose


1 sessions, 1 clients
Cumulative transmit rate 1.2 pps, cumulative receive rate 1.2 pps
```

show bfd session summary

```
user@host> show bfd session summary
2 sessions, 2 clients
Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

PART 4

Index

Index