Tunnel and Encryption Services Interfaces Feature Guide for Routing Devices

Published: 2014-09-27

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

Tunnel and Encryption Services Interfaces Feature Guide for Routing Devices

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:

- M Series
- MX Series
- T 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 xiii 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 xiv defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

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

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

Convention	Description	Examples
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**.
> (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

Tunnel Services

CHAPTER 1

Overview

Tunnel Services Overview

By encapsulating arbitrary packets inside a transport protocol, tunneling provides a private, secure path through an otherwise public network. Tunnels connect discontinuous subnetworks and enable encryption interfaces, virtual private networks (VPNs), and MPLS. If you have a Tunnel Physical Interface Card (PIC) installed in your M Series or T Series router, you can configure unicast, multicast, and logical tunnels.

You can configure two types of tunnels for VPNs: one to facilitate routing table lookups and another to facilitate VPN routing and forwarding instance (VRF) table lookups.

For information about encryption interfaces, see "Configuring Encryption Interfaces" on page 55 and the *Junos OS Administration Library for Routing Devices*. For information about VPNs, see the *Junos OS VPNs Library for Routing Devices*. For information about MPLS, see the *Junos OS MPLS Applications Library for Routing Devices*.

On SRX Series devices, Generic Routing Encapsulation (GRE) and IP-IP tunnels use internal interfaces, gr-0/0/0 and ip-0/0/0, respectively. The Junos OS creates these interfaces at system bootup; they are not associated with physical interfaces.

The Juniper Networks Junos OS supports the tunnel types shown in Table 3 on page 3.

Table 3: Tunnel Interface Types

Interface	Description
gr-0/0/0	Configurable generic routing encapsulation (GRE) interface. GRE allows the encapsulation of one routing protocol over another routing protocol. Within a router, packets are routed to this internal interface, where they are first encapsulated with a GRE packet and then re-encapsulated with another protocol packet to complete the GRE. The GRE interface is an internal interface only and is not associated with a physical interface. You must configure the interface for it to perform GRE.

Table 3: Tunnel Interface Types *(continued)*

Interface	Description
gre	Internally generated GRE interface. This interface is generated by the Junos OS to handle GRE. NOTE: You can configure GRE interfaces (gre-x/y/z) only for GMPLS control channels. GRE interfaces are not supported or configurable for other applications. This type of interface does not require a Tunnel PIC. For more information about GMPLS, see the *Junos OS MPLS Applications Library for Routing Devices* and the *Junos OS, Release 14.1*.
ip-0/0/0	Configurable IP-over-IP encapsulation (also called IP tunneling) interface. IP tunneling allows the encapsulation of one IP packet over another IP packet. Packets are routed to an internal interface where they are encapsulated with an IP packet and then forwarded to the encapsulating packet's destination address. The IP-IP interface is an internal interface only and is not associated with a physical interface. You must configure the interface for it to perform IP tunneling.
ipip	Internally generated IP-over-IP interface. This interface is generated by the Junos OS to handle IP-over-IP encapsulation. It is not a configurable interface.
lt-0/0/0	The **lt** interface on M Series and T Series routers supports configuration of logical systems—the capability to partition a single physical router into multiple logical devices that perform independent routing tasks. On SRX Series devices, the **lt** interface is a configurable logical tunnel interface that interconnects logical systems. See the *Junos OS Logical Systems Configuration Guide for Security Devices*.
mt-0/0/0	Internally generated multicast tunnel interface. Multicast tunnels filter all unicast packets; if an incoming packet is not destined for a **224/8**-or-greater prefix, the packet is dropped and a counter is incremented. Within a router, packets are routed to this internal interface for multicast filtering. The multicast tunnel interface is an internal interface only and is not associated with a physical interface. If your router has a Tunnel Services PIC, the Junos OS automatically configures one multicast tunnel interface (**mt-**) for each virtual private network (VPN) you configure. You do not need to configure multicast tunnel interfaces. However, you can configure properties on **mt-** interfaces, such as the **multicast-only** statement.
mtun	Internally generated multicast tunnel interface. This interface is generated by the Junos OS to handle multicast tunnel services. It is not a configurable interface.

Table 3: Tunnel Interface Types *(continued)*

Interface	Description
pd-0/0/0	Configurable Protocol Independent Multicast (PIM) de-encapsulation interface. In PIM sparse mode, the first-hop router encapsulates packets destined for the rendezvous point router. The packets are encapsulated with a unicast header and are forwarded through a unicast tunnel to the rendezvous point. The rendezvous point then de-encapsulates the packets and transmits them through its multicast tree. Within a router, packets are routed to this internal interface for de-encapsulation. The PIM de-encapsulation interface is an internal interface only and is not associated with a physical interface. You must configure the interface for it to perform PIM de-encapsulation. NOTE: On SRX Series devices, this interface type is **ppd0**.
pe-0/0/0	Configurable PIM encapsulation interface. In PIM sparse mode, the first-hop router encapsulates packets destined for the rendezvous point router. The packets are encapsulated with a unicast header and are forwarded through a unicast tunnel to the rendezvous point. The rendezvous point then de-encapsulates the packets and transmits them through its multicast tree. Within a router, packets are routed to this internal interface for encapsulation. The PIM encapsulation interface is an internal interface only and is not associated with a physical interface. You must configure the interface for it to perform PIM encapsulation. NOTE: On SRX Series devices, this interface type is **ppe0**.
pimd	Internally generated PIM de-encapsulation interface. This interface is generated by the Junos OS to handle PIM de-encapsulation. It is not a configurable interface.
pime	Internally generated PIM encapsulation interface. This interface is generated by the Junos OS to handle PIM encapsulation. It is not a configurable interface.
vt-0/0/0	Configurable virtual loopback tunnel interface. Facilitates VRF table lookup based on MPLS labels. This interface type is supported on M Series and T Series routers, but not on SRX Series devices. To configure a virtual loopback tunnel to facilitate VRF table lookup based on MPLS labels, you specify a virtual loopback tunnel interface name and associate it with a routing instance that belongs to a particular routing table. The packet loops back through the virtual loopback tunnel for route lookup.

Related Documentation

- GRE Keepalive Time Overview on page 9
- Configuring Unicast Tunnels on page 17
- Restricting Tunnels to Multicast Traffic on page 23
- Configuring Tunnel Interfaces on MX Series Routers on page 6
- Configuring Tunnel Interfaces on T4000 Routers on page 7

Configuring Tunnel Interfaces on MX Series Routers

Because the MX Series routers do not support Tunnel Services PICs, you create tunnel interfaces on MX Series routers by including the following statements at the **[edit chassis]** hierarchy level:

```
[edit chassis]
fpc slot-number {
  pic number {
    tunnel-services {
      bandwidth (1g | 10g | 20g | 40g);
    }
  }
}
```

fpc *slot-number* is the slot number of the DPC, MPC, or MIC. On the MX80 router, the range is 0 through 1. On other MX series routers, if two SCBs are installed, the range is 0 through 11. If three SCBs are installed, the range is 0 through 5 and 7 through 11.

The **pic *number*** On MX80 routers, if the FPC is 0, the PIC number can only be 0. If the FPC is 1, the PIC range is 0 through 3. For all other MX series routers, the range is 0 through 3.

bandwidth (**1g** | **10g** | **20g** | **40g**) is the amount of bandwidth to reserve for tunnel traffic on each Packet Forwarding Engine.

NOTE: When you use MPCs and MICs, tunnel interfaces are soft interfaces and allow as much traffic as the forwarding-path allows, so it is advantageous to setup tunnel services without artificially limiting traffic by use of the bandwidth option. However, you *must* specify bandwidth when configuring tunnel services for MX Series routers with DPCs or FPCs. The GRE key option is not supported on the tunnel interfaces for DPCs on MX960 routers.

Bandwidth rates of 20 gigabits per second and 40 gigabits per second require use of an MX Series router with the 100-Gigabit Ethernet Modular Port Concentrator (MPC) and the 100-Gigabit CFP MIC.

1g indicates that 1 gigabit per second of bandwidth is reserved for tunnel traffic.

10g indicates that 10 gigabits per second of bandwidth is reserved for tunnel traffic.

20g indicates that 20 gigabits per second of bandwidth is reserved for tunnel traffic.

40g indicates that 40 gigabits per second of bandwidth is reserved for tunnel traffic.

If you specify a bandwidth that is not compatible, tunnel services are not activated. For example, you cannot specify a bandwidth of 1 Gbps for a Packet Forwarding Engine on a 10-Gigabit Ethernet 4-port DPC.

To verify that the tunnel interfaces have been created, issue the **show interfaces terse** operational mode command. For more information, see the CLI Explorer. The bandwidth

that you specify determines the port number of the tunnel interfaces that are created. When you specify a bandwidth of **1g**, the port number is always 10. When you specify any other bandwidth, the port number is always 0.

NOTE: Ingress queueing and tunnel services cannot be configured on the same MPC as it causes PFE forwarding to stop. Each feature can, however, be configured and used separately.

Related Documentation

- *Example: Configuring Tunnel Interfaces on a Gigabit Ethernet 40-Port DPC*
- *Example: Configuring Tunnel Interfaces on a 10-Gigabit Ethernet 4-Port DPC*
- *Example: Configuring Tunnel Interfaces on the MPC3E*
- *bandwidth (Tunnel Services)*
- *tunnel-services (Chassis)*
- *[edit chassis] Hierarchy Level*

Configuring Tunnel Interfaces on T4000 Routers

To create tunnel interfaces on a T4000 Core Router, include the following statements at the **[edit chassis]** hierarchy level:

```
[edit chassis]
fpc slot-number {
  pic number {
    tunnel-services {
      bandwidth bandwidth-value;
    }
  }
}
```

fpc *slot-number* denotes the slot number of the FPC. On the T4000 router, the range is 0 through 7.

NOTE:

- This applies only to the T4000 Type 5 FPC. If any other type of FPC is configured in this slot, this configuration is ignored and no tunnel physical interface is created.
- When you use Type 5 FPCs, the tunnel interfaces are soft interfaces and allow as much traffic as the forwarding-path allows. So, it is advantageous to setup tunnel services without artificially limiting traffic by setting the **bandwidth** statement.

pic *number* on the T4000 router is 0 or 1.

bandwidth *bandwidth-value* is the amount of bandwidth to reserve for the tunnel traffic on each Packet Forwarding Engine. The bandwidth value accepted includes every multiple of 10g up to 100g.

If you specify a bandwidth that is not compatible, tunnel services are not activated. For example, you cannot specify a bandwidth of 1 Gbps for a Packet Forwarding Engine on a 100-Gigabit Ethernet PIC with CFP.

To verify that the tunnel interfaces have been created, issue the **show interfaces terse** operational mode command. For more information, see the *Junos Interfaces Command Reference*.

Related Documentation

- *bandwidth (Tunnel Services)*
- *tunnel-services (Chassis)*
- *[edit chassis] Hierarchy Level*

CHAPTER 2

Encapsulating One Protocol Over Another Using GRE Interfaces

GRE Keepalive Time Overview

Generic routing encapsulation (GRE) tunnel interfaces do not have a built-in mechanism for detecting when a tunnel is down. You can enable keepalive messages to serve as the detection mechanism.

Keepalives can be configured on the physical or on the logical interface. If configured on the physical interface, keepalives are sent on all logical interfaces that are part of the physical interface. If configured on a individual logical interface, keepalives are only sent to that logical interface. In addition to configuring a keepalive, you must configure the hold time.

Related Documentation

- Configuring GRE Keepalive Time on page 9
- **keepalive-time on page 82**
- **hold-time on page 80**

Configuring GRE Keepalive Time

Configuring Keepalive Time and Hold time for a GRE Tunnel Interface

You can configure the keepalives on a generic routing encapsulation (GRE) tunnel interface by including both the **keepalive-time** statement and the **hold-time** statement at the **[edit protocols oam gre-tunnel interface *interface-name*]** hierarchy level.

NOTE: For proper operation of keepalives on a GRE interface, you must also include the **family inet** statement at the **[edit interfaces *interface-name* unit *unit*]** hierarchy level. If you do not include this statement, the interface is marked as down.

To configure a GRE tunnel interface:

1. Configure the GRE tunnel interface at **[edit interfaces *interface-name* unit *unit-number*]** hierarchy level, where the interface name is gr-x/y/z, and the family is set as **inet**.

   user@host# **set interfaces *interface-name* unit *unit-number* family *family-name***

2. Configure the rest of the GRE tunnel interface options as explained in *Configuring a GRE Tunnel Interface Between a PE and CE Router* or *Configuring a GRE Tunnel Interface Between PE Routers* based on requirement.

To configure keepalive time for a GRE tunnel interface:

1. Configure the Operation, Administration, and Maintenance (OAM) protocol at the **[edit protocols]** hierarchy level for the GRE tunnel interface.

   [edit]
   user@host# **edit protocols oam**

2. Configure the GRE tunnel interface option for OAM protocol.

   [edit protocols oam]
   user@host# **edit gre-tunnel interface *interface-name***

3. Configure the keepalive time from 1 through 50 seconds for the GRE tunnel interface.

   [edit protocols oam gre-tunnel interface *interface-name*]
   user@host# **set** keepalive-time ***seconds***

4. Configure the hold time from 5 through 250 seconds. Note that the hold time must be at least twice the keepalive time.

   [edit protocols oam gre-tunnel interface *interface-name*]
   user@host# **set** hold-time ***seconds***

Display GRE Keepalive Time Configuration

Purpose Display the configured keepalive time value as 10 and hold time value as 30 on a GRE tunnel interface (for example, gr-1/1/10.1).

Action To display the configured values on the GRE tunnel interface, run the **show oam gre-tunnel** command at the **[edit protocols]** hierarchy level:

```
[edit protocols]
user@host# show oam gre-tunnel
  interface gr-1/1/10.1 {
    keepalive-time 10;
    hold-time 30;
  }
```

Display Keepalive Time Information on a GRE Tunnel Interface

Purpose Display the current status information of a GRE tunnel interface when keepalive time and hold time parameters are configured on it and when the hold time expires.

Action To verify the current status information on a GRE tunnel interface (for example, gr-3/3/0.3), run the **show interfaces gr-3/3/0.3 terse** and **show interfaces gr-3/3/0.3 extensive** operational commands.

show interfaces gr-3/3/0.3 terse

```
user@host> show interfaces gr-3/3/0.3 terse

Interface               Admin Link Proto    Local                 Remote
gr-3/3/0.3              up    up   inet     200.1.3.1/24
                                  mpls
```

show interfaces gr-3/3/0.3 extensive

```
user@host> show interfaces gr-3/3/0.3 extensive
Logical interface gr-3/3/0.3 (Index 73) (SNMP ifIndex 594) (Generation 900)
    Flags: Point-To-Point SNMP-Traps 0x4000 IP-Header
10.1.19.11:10.1.19.12:47:df:64:0000000000000000 Encapsulation: GRE-NULL
    Gre keepalives configured: On, Gre keepalives adjacency state: down
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
    Traffic statistics:
     Input  bytes  :             15629992
     Output bytes  :             15912273
     Input  packets:               243813
     Output packets:               179476
    Local statistics:
     Input  bytes  :             15322586
     Output bytes  :             15621359
     Input  packets:               238890
     Output packets:               174767
    Transit statistics:
     Input  bytes  :               307406                    0 bps
     Output bytes  :               290914                    0 bps
     Input  packets:                 4923                    0 pps
     Output packets:                 4709                    0 pps
    Protocol inet, MTU: 1476, Generation: 1564, Route table: 0
      Flags: Sendbcast-pkt-to-re
      Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
        Destination: 200.1.3/24, Local: 200.1.3.1, Broadcast: 200.1.3.255,
Generation: 1366
    Protocol mpls, MTU: 1464, Maximum labels: 3, Generation: 1565, Route table:
0
```

NOTE:
When the hold time expires:

- The GRE tunnel will stay up even though the interface cannot send or receive traffic.
- The Link status will be Up and the Gre keepalives adjacency state will be Down.

Meaning The current status information of a GRE tunnel interface with keepalive time and hold time parameters is displayed as expected when the hold time expires.

Related Documentation

- GRE Keepalive Time Overview on page 9
- keepalive-time on page 82
- hold-time on page 80

Enabling Fragmentation on GRE Tunnels

To enable fragmentation of IPv4 packets in generic routing encapsulation (GRE) tunnels, include the **clear-dont-fragment-bit** statement and a maximum transmission unit (MTU) setting for the tunnel as part of an existing GRE configuration at the **[edit interfaces]** hierarchy level:

```
[edit interfaces]
gr-fpc/pic/port {
    unit logical-unit-number {
        clear-dont-fragment-bit;
        ...
        family inet {
            mtu 1000;
            ...
        }
    }
}
```

This statement clears the Don't Fragment (DF) bit in the packet header, regardless of the packet size. If the packet size exceeds the tunnel MTU value, the packet is fragmented before encapsulation. The maximum MTU size configurable on the AS or Multiservices PIC is 9192 bytes.

NOTE: The clear-dont-fragment-bit statement is supported only on MX Series routers and all M Series routers except the M320 router.

Fragmentation is enabled only on IPv4 packets being encapsulated in IPv4-based GRE tunnels.

NOTE: This configuration is supported only on GRE tunnels on AS or Multiservices interfaces. If you commit **gre-fragmentation** as the encapsulation type on a standard Tunnel PIC interface, the following console log message appears when the PIC comes online:

gr-*fpc/pic/port*: does not support this encapsulation

The Packet Forwarding Engine updates the IP identification field in the outer IP header of GRE-encapsulated packets, so that reassembly of the packets is possible after fragmentation. The previous CLI constraint check that required you to configure either the **clear-dont-fragment-bit** statement or a tunnel key with the **allow-fragmentation** statement is no longer enforced.

When you configure the **clear-dont-fragment-bit** statement on an interface with the MPLS protocol family enabled, you must specify an MTU value. This MTU value must not be greater than maximum supported value (9192).

Related Documentation

- Configuring Unicast Tunnels on page 17

CHAPTER 3

Encapsulating One IP Packet Over Another Using IP-IP Interfaces

Configuring IPv6-over-IPv4 Tunnels

If you have a Tunnel PIC installed in your M Series or T Series router, you can configure IPv6-over-IPv4 tunnels. To define a tunnel, you configure a unicast tunnel across an existing IPv4 network infrastructure. IPv6/IPv4 packets are encapsulated in IPv4 headers and sent across the IPv4 infrastructure through the configured tunnel. You manually configure configured tunnels on each end point.

On SRX Series devices, Generic Routing Encapsulation (GRE) and IP-IP tunnels use internal interfaces, gr-0/0/0 and ip-0/0/0, respectively. The Junos OS creates these interfaces at system bootup; they are not associated with a physical interface.

IPv6-over-IPv4 tunnels are defined in RFC 2893, *Transition Mechanisms for IPv6 Hosts and Routers*. For information about configuring a unicast tunnel, see "Configuring Unicast Tunnels" on page 17. For an IPv6-over-IPv4 tunnel configuration example, see "Example: Configuring an IPv6-over-IPv4 Tunnel" on page 15.

Related Documentation

- Tunnel Services Overview on page 3
- Example: Configuring an IPv6-over-IPv4 Tunnel on page 15

Example: Configuring an IPv6-over-IPv4 Tunnel

Configure a tunnel on both sides of the connection.

Configuration on Router 1

```
[edit]
interfaces {
   gr-1/0/0 {
      unit 0 {
         tunnel {
            source 10.19.2.1;
            destination 10.19.3.1;
         }
```

```
        family inet6 {
          address 2001:DB8:1:1/126;
        }
      }
    }
  }
```

Configuration on Router 2

```
[edit]
interfaces {
  gr-1/0/0 {
    unit 0 {
      tunnel {
        source 10.19.3.1;
        destination 10.19.2.1;
      }
      family inet6 {
        address 2001:DB8:2:1/126;
      }
    }
  }
}
```

Related Documentation

- Tunnel Services Overview on page 3
- Configuring IPv6-over-IPv4 Tunnels on page 15

CHAPTER 4

Filtering Unicast Packets Through Multicast Tunnel Interfaces

Configuring Unicast Tunnels

To configure a unicast tunnel, you configure a **gr-** interface (to use GRE encapsulation) or an **ip-** interface (to use IP-IP encapsulation) and include the **tunnel** and **family** statements:

```
gr-fpc/pic/port or ip-fpc/pic/port {
    unit logical-unit-number {
        copy-tos-to-outer-ip-header;
        reassemble-packets;
        tunnel {
            allow-fragmentation;
            backup-destination address;
            destination destination-address;
            do-not-fragment;
            key number;
            routing-instance {
                destination routing-instance-name;
            }
            source address;
            ttl number;
        }
        family family {
            address address {
                destination address;
            }
        }
    }
}
```

You can configure these statements at the following hierarchy levels:

- **[edit interfaces]**

- **[edit logical-systems *logical-system-name* interfaces]**

You can configure multiple logical units for each GRE or IP-IP interface, and you can configure only one tunnel per unit.

NOTE: On M Series and T Series routers, you can configure the interface on a service PIC or a tunnel PIC. On MX Series routers, configure the interface on a Multiservices DPC.

Each tunnel interface must be a point-to-point interface. Point to point is the default interface connection type, so you do not need to include the **point-to-point** statement in the logical interface configuration.

You must specify the tunnel's destination and source addresses. The remaining statements are optional.

NOTE: For transit packets exiting the tunnel, forwarding path features, such as reverse path forwarding (RPF), forwarding table filtering, source class usage, destination class usage, and stateless firewall filtering, are not supported on the interfaces you configure as tunnel sources, but are supported on tunnel-pic interfaces.

However, class-of-service (CoS) information obtained from the GRE or IP-IP header is carried over the tunnel and is used by the re-entering packets. For more information, see the *Class of Service Feature Guide for Routing Devices*.

To prevent an invalid configuration, the Junos OS disallows setting the address specified by the **source** or **destination** statement at the **[edit interfaces gr-*fpc/pic/port* unit *logical-unit-number* tunnel]** hierarchy level to be the same as the interface's own subnet address, specified by the **address** statement at the **[edit interfaces gr-*fpc/pic/port* unit *logical-unit-number* family *family-name*]** hierarchy level.

To set the time-to-live (TTL) field that is included in the encapsulating header, include the **ttl** statement. If you explicitly configure a TTL value for the tunnel, you must configure it to be one larger than the number of hops in the tunnel. For example, if the tunnel has seven hops, you must configure a TTL value of 8.

You must configure at least one family on the logical interface. To enable MPLS over GRE tunnel interfaces, you must include the **family mpls** statement in the GRE interface configuration. In addition, you must include the appropriate statements at the **[edit protocols]** hierarchy level to enable Resource Reservation Protocol (RSVP), MPLS, and label-switched paths (LSPs) over GRE tunnels. Unicast tunnels are bidirectional.

A configured tunnel cannot go through Network Address Translation (NAT) at any point along the way to the destination. For more information, see "Examples: Configuring Unicast Tunnels" on page 22 and the *Junos OS MPLS Applications Library for Routing Devices*.

For a GRE tunnel, the default is to set the ToS bits in the outer IP header to all zeros. To have the Routing Engine copy the ToS bits from the inner IP header to the outer, include the **copy-tos-bits-to-outer-ip-header** statement. (This inner-to-outer ToS bits copying is already the default behavior for IP-IP tunnels.)

For GRE tunnel interfaces on Adaptive Services or Multiservices interfaces, you can configure additional tunnel attributes, as described in the following sections:

Configuring a Key Number on GRE Tunnels

For Adaptive Services and Multiservices interfaces on M Series and T Series routers, you can assign a key value to identify an individual traffic flow within a GRE tunnel, as defined in RFC 2890, *Key and Sequence Number Extensions to GRE*. However, only one key is allowed for each tunnel source and destination pair.

Each IP version 4 (IPv4) packet entering the tunnel is encapsulated with the GRE tunnel key value. Each IPv4 packet exiting the tunnel is verified by the GRE tunnel key value and de-encapsulated. The Adaptive Services or Multiservices PIC drops packets that do not match the configured key value.

To assign a key value to a GRE tunnel interface, include the **key** statement:

```
key number;
```

You can include this statement at the following hierarchy levels:

- **[edit interfaces *interface-name* unit *logical-unit-number* tunnel]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* tunnel]**

The key number can be 0 through 4,294,967,295. You must configure the same GRE tunnel key value on tunnel endpoints.

The following example illustrates the use of the key statement in a GRE tunnel configuration:

```
interfaces {
   gr-1/2/0 {
      unit 0 {
         tunnel {
            source 10.58.255.193;
            destination 10.58.255.195;
            key 1234;
         }
         ...
         family inet {
```

```
            mtu 1500;
            address 10.200.0.1/30;
            ...
        }
    }
}
}
```

Enabling Fragmentation on GRE Tunnels

For GRE tunnel interfaces on Adaptive Services and Multiservices interfaces only, you can enable fragmentation of IPv4 packets in GRE tunnels.

By default, IPv4 traffic transmitted over GRE tunnels is not fragmented. To enable fragmentation of IPv4 packets in GRE tunnels, include the **clear-dont-fragment-bit** statement:

```
clear-dont-fragment-bit;
```

You can include this statement at the following hierarchy levels:

- **[edit interfaces** *interface-name* **unit** *logical-unit-number***]**
- **[edit logical-systems** *logical-system-name* **interfaces** *interface-name* **unit** *logical-unit-number***]**

When you include the **clear-dont-fragment-bit** statement in the configuration, the don't-fragment (DF) bit is cleared on all packets, even packets that do not exceed the tunnel maximum transmission unit (MTU). If the packet's size exceeds the tunnel's MTU value, the packet is fragmented before encapsulation. If the packet's size does not exceed the tunnel's MTU value, the packet is not fragmented.

NOTE: The Packet Forwarding Engine updates the IP identification field in the outer IP header of GRE-encapsulated packets, so that reassembly of the packets is possible after fragmentation. The previous CLI constraint check that required you to configure either the **clear-dont-fragment-bit** statement or a tunnel key with the **allow-fragmentation** statement is no longer enforced.

You can also clear the DF bit in packets transmitted over IP Security (IPsec) tunnels. For more information, see *Enabling IPsec Packet Fragmentation*.

Specifying an MTU Setting for the Tunnel

To enable key numbers and fragmentation on GRE tunnels (as described in "Configuring a Key Number on GRE Tunnels" on page 19 and "Enabling Fragmentation on GRE Tunnels" on page 20), you must also specify an MTU setting for the tunnel.

To specify an MTU setting for the tunnel, include the **mtu** statement:

```
mtu bytes;
```

You can include this statement at the following hierarchy levels:

- **[edit interfaces gr-*fpc/pic/port* unit *logical-unit-number* family inet]**
- **[edit logical-system *logical-system-name* interfaces gr-*fpc/pic/port* unit *logical-unit-number* family inet]**

For more information about MTU settings, see the *Junos OS Network Interfaces Library for Routing Devices*.

Configuring a GRE Tunnel to Copy ToS Bits to the Outer IP Header

Unlike IP-IP tunnels, GRE tunnels do not copy the ToS bits to the outer IP header by default. To have the Routing Engine copy the inner ToS bits to the outer IP header (which is required for some tunneled routing protocols) on packets sent by the Routing Engine, include the **copy-tos-to-outer-ip-header** statement at the logical unit hierarchy level of a GRE interface. This example copies the inner ToS bits to the outer IP header on a GRE tunnel:

```
[edit interfaces]
gr-0/0/0 {
  unit 0 {
    copy-tos-to-outer-ip-header;
    family inet;
  }
}
```

Configuring Packet Reassembly

On GRE tunnel interfaces only, you can enable reassembly of fragmented tunnel packets. To activate this capability, include the **reassemble-packets** statement:

```
reassemble-packets;
```

You can configure this statement at the following hierarchy levels:

- **[edit interfaces *interface-name* unit *logical-unit-number*]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]**

For each tunnel you configure on the interface, you can enable or disable fragmentation of GRE packets by including the **allow-fragmentation** or **do-not-fragment** statement:

```
allow-fragmentation;
do-not-fragment;
```

You can configure these statements at the following hierarchy levels:

- **[edit interfaces *interface-name* unit *logical-unit-number* tunnel]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* tunnel]**

If you configure **allow-fragmentation** on a tunnel, it clears the DF bit in the outer IP header, enabling post fragmentation of GRE-encapsulated packets if the packet size exceeds the maximum transmission unit (MTU) value for the egress interface. By default, packets that exceed the MTU size are dropped and post fragmentation of GRE packets is disabled.

NOTE: Whenever you configure **allow-fragmentation** on a tunnel, you must also include either the **tunnel key** or the **clear-dont-fragment-bit** statement. This configuration enables the router to send affected packets to the PIC so that the correct IP header can be placed in the fragments. Otherwise, on the reassembly side some packets might be lost when fragments arrive in the PIC out of sequence at high speeds.

Related Documentation

- Tunnel Services Overview on page 3
- Examples: Configuring Unicast Tunnels on page 22

Examples: Configuring Unicast Tunnels

Configure two unnumbered IP-IP tunnels:

```
[edit interfaces]
ip-0/3/0 {
  unit 0 {
    tunnel {
      source 192.168.4.18;
      destination 192.168.4.253;
    }
    family inet;
  }
  unit 1 {
    tunnel {
      source 192.168.4.18;
      destination 192.168.4.254;
    }
    family inet;
  }
}
```

Configure numbered tunnel interfaces by including an address at the **[edit interfaces ip-0/3/0 unit (0 | 1) family inet]** hierarchy level:

```
[edit interfaces]
ip-0/3/0 {
  unit 0 {
    tunnel {
      source 192.168.4.18;
      destination 192.168.4.253;
    }
    family inet {
      address 10.5.5.1/30;
    }
  }
  unit 1 {
    tunnel {
      source 192.168.4.18;
      destination 192.168.4.254;
    }
```

```
    family inet {
      address 10.6.6.100/30;
    }
  }
}
```

Configure an MPLS over GRE tunnel by including the **family mpls** statement at the **[edit interfaces gr-1/2/0 unit 0]** hierarchy level:

```
[edit interfaces]
gr-1/2/0 {
  unit 0 {
    tunnel {
      source 192.168.1.1;
      destination 192.168.1.2;
    }
    family inet {
      address 10.1.1.1/30;
    }
    family mpls;
  }
}
```

Related Documentation

- Tunnel Services Overview on page 3
- Configuring Unicast Tunnels on page 17

Restricting Tunnels to Multicast Traffic

For interfaces that carry IPv4 or IP version 6 (IPv6) traffic, you can configure a tunnel interface to allow multicast traffic only. To configure a multicast-only tunnel, include the **multicast-only** statement:

```
multicast-only;
```

You can configure this statement at the following hierarchy levels:

- **[edit interfaces** *interface-name* **unit** *logical-unit-number* **family** *family***]**
- **[edit logical-systems** *logical-system-name* **interfaces** *interface-name* **unit** *logical-unit-number* **family** *family***]**

Multicast tunnels filter all unicast packets; if an incoming packet is not destined for a 224/8 or greater prefix, the packet is dropped and a counter is incremented.

You can configure this property on GRE, IP-IP, PIM, and multicast tunnel (**mt**) interfaces only.

NOTE: If your router has a Tunnel Services PIC, the Junos OS automatically configures one multicast tunnel interface (**mt**) for each virtual private network (VPN) you configure. You do not need to configure multicast tunnel interfaces.

Related Documentation

- Tunnel Services Overview on page 3
- Configuring Unicast Tunnels on page 17

CHAPTER 5

Connecting Logical Systems Using Logical Tunnel Interfaces

Configuring Logical Tunnel Interfaces

Logical tunnel (**lt-**) interfaces provide quite different services depending on the host router:

- On M Series, MX Series, and T Series routers, logical tunnel interfaces allow you to connect logical systems, virtual routers, or VPN instances. M Series and T Series routers must be equipped with a Tunnel Services PIC or an Adaptive Services Module (only available on M7i routers). MX Series routers must be equipped with a Trio MPC/MIC module. For more information about connecting these applications, see the *Junos OS VPNs Library for Routing Devices*.
- On SRX Series Services Gateways, the logical tunnel interface is used to interconnect logical systems. See the *Junos OS Logical Systems Configuration Guide for Security Devices*.

For M Series, MX Series, and T Series routers, see the following section:

- Connecting Logical Systems on page 25

Connecting Logical Systems

To connect two logical systems, you configure a logical tunnel interface on both logical systems. Then you configure a peer relationship between the logical tunnel interfaces, thus creating a point-to-point connection.

To configure a point-to-point connection between two logical systems, configure the logical tunnel interface by including the **lt-*fpc/pic/port*** statement:

lt-*fpc/pic/port* {

```
    unit logical-unit-number {
      encapsulation encapsulation;
      peer-unit unit-number; # peering logical system unit number
      dlci dlci-number;
      family (inet | inet6 | iso | mpls);
    }
  }
```

You can include this statement at the following hierarchy levels:

- **[edit interfaces]**
- **[edit logical-systems *logical-system-name* interfaces]**

When configuring logical tunnel interfaces, note the following:

- You can configure each logical tunnel interface with one of the following encapsulation types: Ethernet, Ethernet circuit cross-connect (CCC), Ethernet VPLS, Frame Relay, Frame Relay CCC, VLAN, VLAN CCC, or VLAN VPLS.
- You can configure the IP, IPv6, International Organization for Standardization (ISO), or MPLS protocol family.
- The peering logical interfaces must belong to the same logical tunnel interface derived from the Tunnel Services PIC or Adaptive Services Module.
- You can configure only one peer unit for each logical interface. For example, unit 0 cannot peer with both unit 1 and unit 2.
- To enable the logical tunnel interface, you must configure at least one physical interface statement.
- Logical tunnels are not supported with Adaptive Services, Multiservices, or Link Services PICs (but they are supported on the Adaptive Services Module on M7i routers, as noted above).
- On M Series routers other than the M40e router, logical tunnel interfaces require an Enhanced Flexible PIC Concentrator (FPC).
- On MX Series routers, logical tunnel interfaces require Trio MPC/MIC modules. They do not require a Tunnel Services PIC in the same system.

For more information about configuring logical systems, see the *Junos OS Routing Protocols Library for Routing Devices*.

Related Documentation

- Tunnel Services Overview on page 3
- Example: Configuring Logical Tunnels on page 26

Example: Configuring Logical Tunnels

Configure three logical tunnels:

```
[edit interfaces]
lt-4/2/0 {
  description “Logical tunnel interface connects three logical systems”;
```

```
}
[edit logical-systems]
lr1 {
    interfaces lt-4/2/0 {
        unit 12 {
            peer-unit 21; #Peering with lr2
            encapsulation frame-relay;
            dlci 612;
            family inet;
        }
        unit 13 {
            peer-unit 31; #Peering with lr3
            encapsulation frame-relay-ccc;
            dlci 613;
        }
    }
}
lr2 {
    interfaces lt-4/2/0 {
        unit 21 {
            peer-unit 12; #Peering with lr1
            encapsulation frame-relay-ccc;
            dlci 612;
        }
        unit 23 {
            peer-unit 32; #Peering with lr3
            encapsulation frame-relay;
            dlci 623;
        }
    }
}
lr3 {
    interfaces lt-4/2/0 {
        unit 31 {
            peer-unit 13; #Peering with lr1
            encapsulation frame-relay;
            dlci 613;
            family inet;
        }
        unit 32 {
            peer-unit 23; #Peering with lr2
            encapsulation frame-relay-ccc;
            dlci 623;
        }
    }
}
```

Related Documentation

- Tunnel Services Overview on page 3
- Configuring Logical Tunnel Interfaces on page 25

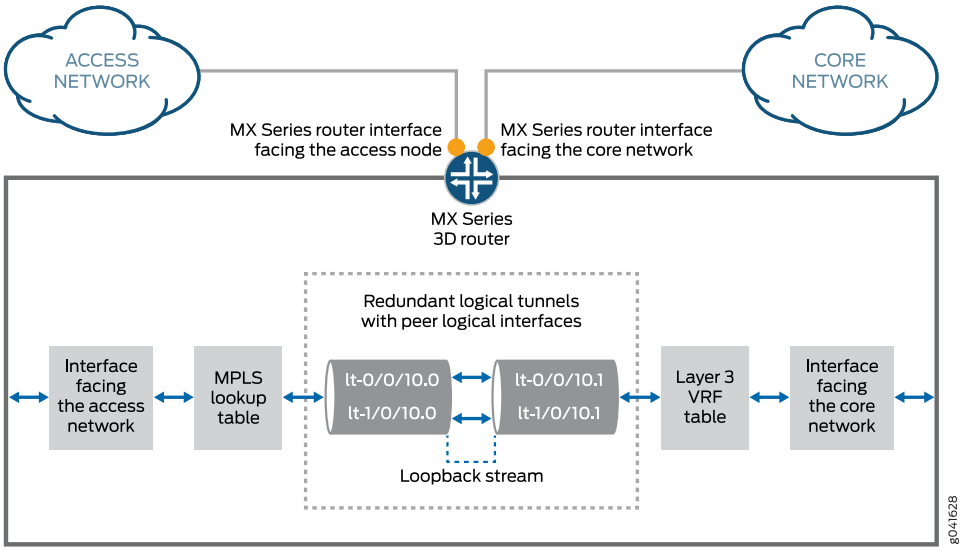
Redundant Logical Tunnels Overview

You can connect two devices, such as an access-facing device and a core-facing device, through logical tunnels. To provide redundancy for the tunnels, you can create and configure multiple physical logical tunnels and add them to a virtual redundant logical tunnel.

NOTE: **Redundant logical tunnels are supported only on MX Series routers with MPCs.**

For example, in an MPLS access network, you can configure multiple pseudowires between an access node and an MX Series router with MPCs and add them to a redundant logical tunnel. You can then add multiple logical tunnels to the redundant logical tunnel. Figure 1 on page 28 shows a redundant logical tunnel between the access node and the MX Series router.

Figure 1: Redundant Logical Tunnels



The redundant logical tunnel has peer logical interfaces at each end, rlt0.0 and rlt0.1. You can configure router features on these interfaces for the redundant logical tunnel and its members.

Each member logical tunnel has peer logical interfaces. In Figure 1 on page 28, lt-0/0/10.0 and lt-0/0/10.1 are peers.

The MX Series router performs IP lookup in the Layer 3 VPN routing and forwarding (VRF) table on the router where the pseudowires that are grouped in logical tunnels terminate.

Redundant Logical Tunnel Configuration

In Junos Releases 13.3R1, 13.3R2, and 14.1R1 you can create up to 16 redundant logical tunnels, depending on the number of Packet Forwarding Engines and the number of

loopback interfaces on each Packet Forwarding Engine on your device. For Junos Release 13.3R3, the valid range for device-count is from 1 to 255.

You can add up to 32 logical tunnels as members of a redundant logical tunnel.

When you add more than two members to the redundant logical tunnel, they are in active mode. The traffic is load-balanced over all the tunnel members.

When you add only two members to the redundant logical tunnel, you can configure the members in one of these ways:

- Both members in active mode
- One member in active mode and the other in backup mode

Redundant Logical Tunnel Failure Detection and Failover

A logical tunnel fails and is removed from the redundant logical tunnel group, and the backup logical tunnel becomes active due to one of these events:

- A hardware failure on the MPC module occurs.
- An MPC failure occurs due to a microkernel crash.
- The MPC module is administratively shut down and removed from the redundant logical tunnel.
- A power failure on the MPC module occurs.

NOTE: You can decrease the time it takes for failure detection and failover to occur. Configure the enhanced-ip statement at the [edit chassis network-services] hierarchy level to enable Packet Forwarding Engine liveliness detection.

Related Documentation

- Example: Configuring Redundant Logical Tunnels on page 31
- *Pseudowire Subscriber Logical Interfaces Overview*
- Configuring Logical Tunnel Interfaces on page 25
- Configuring Redundant Logical Tunnels on page 30
- *Configuring a Pseudowire Subscriber Logical Interface Device*

Configuring Redundant Logical Tunnels

Use redundant logical tunnels to provide redundancy for logical tunnels between two devices, such as an access-facing device and a core-facing device.

When configuring redundant logical tunnel interfaces, note the following:

- In Junos OS Release 13.3 or later, you can configure redundant logical tunnels only on MX Series routers with MPCs.

  In Junos Releases 13.3R1, 13.3R2, and 14.1R1 you can create up to 16 redundant logical tunnels, depending on the number of Packet Forwarding Engines and the number of loopback interfaces on each Packet Forwarding Engine on your device. For Junos Release 13.3R3, the valid range for device-count is from 1 to 255. The command is shown below

  set chassis redundancy-group interface-type redundant-logical-tunnel device-count *[number]*;

  You can add up to 32 logical tunnels as members.
- When a logical tunnel with an existing configuration joins a redundant logical tunnel, you must configure the redundant logical tunnel with the settings from the existing configuration.
- You can add member logical tunnels to a parent logical tunnel for redundancy.
- When you add more than two logical tunnels to the redundant logical tunnel, the members are in active mode by default.
- When you add only two members, you can configure the members in one of these ways:
  - Both members in active mode
  - One member in active mode and the other in backup mode

To configure a redundant logical tunnel between two devices:

1. Create the logical tunnel and redundant logical tunnel interfaces.

   ```
   [edit chassis]
   user@host# set redundancy-group interface-type redundant-logical-tunnel
     device-count count
   user@host# set fpc slot-number pic number tunnel-services bandwidth 1g
   ```

2. Bind the member logical tunnels to the redundant logical tunnel.

   ```
   [edit interfaces]
   user@host# set interface-name redundancy-group member-interface interface-name
   ```

3. Configure the redundant logical tunnel interfaces.
4. Attach the redundant logical tunnel interface to a Layer 2 circuit.
5. Add the peer redundant logical tunnel interface to a Layer 3 VRF instance.
6. Configure MPLS and LDP in the pseudowires and the Layer 3 VPN.

```
[edit protocols]
user@host# set mpls no-cspf
user@host# set mpls interface all
user@host# set ldp interface all
```

7. Configure BGP in the Layer 3 VPN.
8. Configure OSPF on the core-facing interfaces and the router local loopback interface.
9. Set the policy options for BGP.
10. Set the router ID and the autonomous system (AS) number.

Related Documentation

- Example: Configuring Redundant Logical Tunnels on page 31
- Redundant Logical Tunnels Overview on page 28

Example: Configuring Redundant Logical Tunnels

This example shows how to configure redundant logical tunnels in an MPLS access network.

Requirements

In Junos OS Release 13.3 or later, you can configure redundant logical tunnels only on MX Series routers with MPCs.

Overview

When a logical tunnel with an existing configuration joins a redundant logical tunnel, you must configure the redundant logical tunnel with the settings from the existing configuration.

You can add member logical tunnels to a parent logical tunnel for redundancy.

On MX Series routers with MPCs, you can configure redundant logical tunnels as follows:

- In Junos Releases 13.3R1, 13.3R2, and 14.1R1 you can create up to 16 redundant logical tunnels, depending on the number of Packet Forwarding Engines and the number of loopback interfaces on each Packet Forwarding Engine on your device. For Junos Release 13.3R3, the valid range for device-count is from 1 to 255. The command is shown below

  set chassis redundancy-group interface-type redundant-logical-tunnel device-count *[number]*;

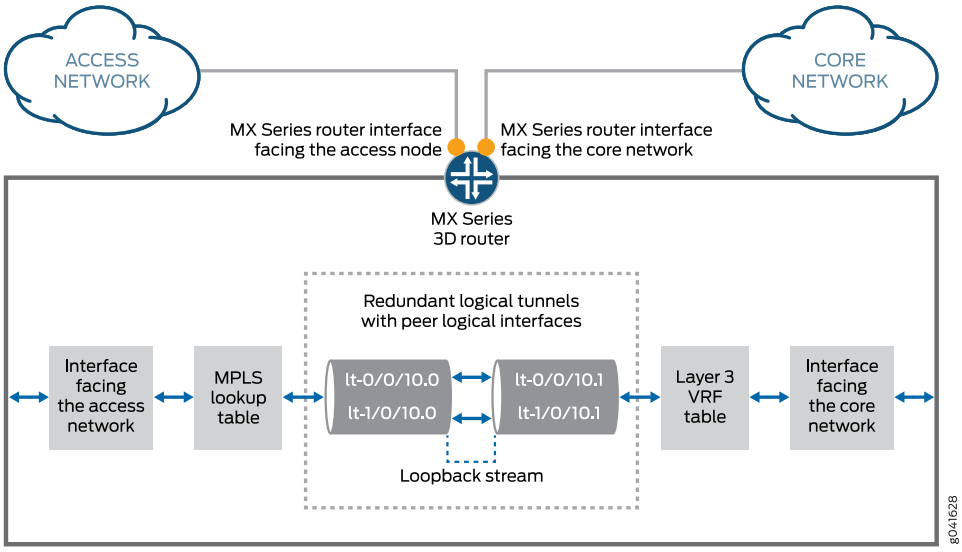
- You can add up to 32 logical tunnels as members.

- When you add more than two logical tunnels to a redundant logical tunnel, the members are in active mode by default.
- When you add only two members, you can configure the members in one of these ways:
  - Both members in active mode
  - One member in active mode and the other in backup mode

Topology

Figure 1 on page 28 shows a redundant logical tunnel between the access node and the MX Series router in an MPLS access network.

Figure 2: Redundant Logical Tunnels



The redundant logical tunnel has peer logical interfaces at each end, rlt0.0 and rlt0.1. You can configure router features on these interfaces for the redundant logical tunnel and its members.

Each member logical tunnel has peer logical interfaces on the access-facing and core-facing devices. In Figure 1 on page 28, lt-0/0/10.0 and lt-0/0/10.1 are peers.

The MX Series router performs IP lookup in the Layer 3 VPN routing and forwarding (VRF) table on the router where the pseudowires that are grouped in logical tunnels terminate.

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.

```
set chassis redundancy-group interface-type redundant-logical-tunnel device-count 4
set chassis fpc 1 pic 0 tunnel-services bandwidth 1g
```

```
set chassis fpc 1 pic 2 tunnel-services bandwidth 1g
set interfaces rlt0 redundancy-group member-interface lt-1/0/10
set interfaces rlt0 redundancy-group member-interface lt-2/0/10
set interfaces rlt0 unit 0 description "Towards Layer 2 Circuit"
set interfaces rlt0 unit 0 encapsulation vlan-ccc
set interfaces rlt0 unit 0 vlan-id 600
set interfaces rlt0 unit 0 peer-unit 1
set interfaces rlt0 unit 0 family ccc
set interfaces rlt0 unit 1 description "Towards Layer 3 VRF"
set interfaces rlt0 unit 1 encapsulation vlan
set interfaces rlt0 unit 1 vlan-id 600
set interfaces rlt0 unit 1 peer-unit 0
set interfaces rlt0 unit 1 family inet address 10.10.10.2/24
set protocols l2circuit neighbor 2.2.2.2 interface rlt0.0 virtual-circuit-id 100
set protocols l2circuit neighbor 2.2.2.2 interface rlt0.0 no-control-word
set routing-instances pe-vrf instance-type vrf
set routing-instances pe-vrf interface rlt0.1
set routing-instances pe-vrf route-distinguisher 65056:1
set routing-instances pe-vrf vrf-import VPN-A-Import
set routing-instances pe-vrf vrf-export VPN-A-Export
set routing-instances pe-vrf vrf-table-label
set routing-instances pe-vrf protocols ospf export VPN-A-Import
set routing-instances pe-vrf protocols ospf area 0.0.0.0 interface rlt0.1
set protocols mpls no-cspf
set protocols mpls interface all
set protocols ldp interface all
set protocols bgp export local-routes
set protocols bgp group internal type internal
set protocols bgp group internal local-address 3.3.3.3
set protocols bgp group internal family inet any
set protocols bgp group internal family inet-vpn unicast
set protocols bgp group internal neighbor 4.4.4.4
set protocols ospf area 0.0.0.0 interface ge-5/3/8.0
set protocols ospf area 0.0.0.0 interface ge-5/2/5.0
set protocols ospf area 0.0.0.0 interface lo0.3 passive
set policy-options policy-statement VPN-A-Export term a then community add VPN-A
set policy-options policy-statement VPN-A-Export term a then accept
set policy-options policy-statement VPN-A-Export term b then reject
set policy-options policy-statement VPN-A-Import term a from protocol bgp
set policy-options policy-statement VPN-A-Import term a from community VPN-A
set policy-options policy-statement VPN-A-Import term a then accept
set policy-options policy-statement VPN-A-Import term b then reject
set policy-options policy-statement local-routes then accept
set policy-options community VPN-A members target:100:100
set routing-options router-id 3.3.3.3
set routing-options autonomous-system 65056
```

Step-by-Step Procedure

In this example, all the logical tunnels are in active mode.

1. Create the logical tunnel and redundant logical tunnel interfaces.

```
[edit chassis]
user@host# set redundancy-group interface-type redundant-logical-tunnel
  device-count 4
user@host# set fpc 1 pic 0 tunnel-services bandwidth 1g
user@host# set fpc 1 pic 2 tunnel-services bandwidth 1g
```

2. Bind the member logical tunnels to the redundant logical tunnel.

```
[edit interfaces]
user@host# set rlt0 redundancy-group member-interface lt-1/0/10
user@host# set rlt0 redundancy-group member-interface lt-2/0/10
```

3. Configure the redundant logical tunnel interfaces.

```
[edit interfaces]
user@host# set rlt0 unit 0 description "Towards Layer 2 Circuit"
user@host# set rlt0 unit 0 encapsulation vlan-ccc
user@host# set rlt0 unit 0 vlan-id 600
user@host# set rlt0 unit 0 peer-unit 1
user@host# set rlt0 unit 0 family ccc


user@host# set rlt0 unit 1 description "Towards Layer 3 VRF"
user@host# set rlt0 unit 1 encapsulation vlan
user@host# set rlt0 unit 1 vlan-id 600
user@host# set rlt0 unit 1 peer-unit 0
user@host# set rlt0 unit 1 family inet address 10.10.10.2/24
```

4. Attach rlt0.0 to a Layer 2 circuit.

```
[edit protocols]
user@host# set l2circuit neighbor 2.2.2.2 interface rlt0.0 virtual-circuit-id 100
user@host# set l2circuit neighbor 2.2.2.2 interface rlt0.0 no-control-word
```

5. Add rlt0.1 to a Layer 3 VRF instance.

```
[edit routing-instances]
user@host# set pe-vrf instance-type vrf
user@host# set pe-vrf interface rlt0.1
user@host# set pe-vrf route-distinguisher 65056:1
user@host# set pe-vrf vrf-import VPN-A-Import
user@host# set pe-vrf vrf-export VPN-A-Export
user@host# set pe-vrf vrf-table-label
user@host# set pe-vrf protocols ospf export VPN-A-Import
user@host# set pe-vrf protocols ospf area 0.0.0.0 interface rlt0.1
```

6. Configure MPLS and LDP in the pseudowires and the Layer 3 VPN.

```
[edit protocols]
user@host# set mpls no-cspf
user@host# set mpls interface all
user@host# set ldp interface all
```

7. Configure BGP in the Layer 3 VPN.

```
[edit protocols]
user@host# set bgp export local-routes
user@host# set bgp group internal type internal
user@host# set bgp group internal local-address 3.3.3.3
user@host# set bgp group internal family inet any
user@host# set bgp group internal family inet-vpn unicast
user@host# set bgp group internal neighbor 4.4.4.4
```

8. Configure OSPF on the core-facing interfaces and the router local loopback interface.

```
[edit protocols]
user@host# set ospf area 0.0.0.0 interface ge-5/3/8.0
```

```
user@host# set ospf area 0.0.0.0 interface ge-5/2/5.0
user@host# set ospf area 0.0.0.0 interface lo0.3 passive
```

9. Set the policy options for BGP.

```
[edit policy-options]
user@host# set policy-statement VPN-A-Export term a then community add VPN-A
user@host# set policy-statement VPN-A-Export term a then accept
user@host# set policy-statement VPN-A-Export term b then reject
user@host# set policy-statement VPN-A-Import term a from protocol bgp
user@host# set policy-statement VPN-A-Import term a from community VPN-A
user@host# set policy-statement VPN-A-Import term a then accept
user@host# set policy-statement VPN-A-Import term b then reject
user@host# set policy-statement local-routes then accept
user@host# set community VPN-A members target:100:100
```

10. Set the router ID and the autonomous system (AS) number.

```
[edit routing-options]
user@host# set router-id 3.3.3.3
user@host# set autonomous-system 65056
```

Results

From configuration mode, confirm your configuration by entering the following commands:

- **show chassis**
- **show interfaces**
- **show policy-options**
- **show protocols**
- **show routing-instances**
- **show routing-options**

If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show chassis
redundancy-group {
   interface-type {
      redundant-logical-tunnel {
         device-count 4;
      }
   }
}
fpc 1 {
   pic 0 {
      tunnel-services {
         bandwidth 1g;
      }
   }
}
fpc 1 {
   pic 2 {
```

```
      tunnel-services {
        bandwidth 1g;
      }
    }
}
```

```
user@host# show interfaces rlt0
redundancy-group {
  member-interface lt-1/0/10;
  member-interface lt-2/0/10;
}
unit 0 {
  description "Towards Layer 2 Circuit";
  encapsulation vlan-ccc;
  vlan-id 600;
  peer-unit 1;
  family ccc;
}
unit 1 {
  description "Towards Layer 3 VRF";
  encapsulation vlan;
  vlan-id 600;
  peer-unit 0;
  family inet {
    address 10.10.10.2/24;
  }
}
```

```
user@host# show protocols l2circuit
neighbor 2.2.2.2 {
  interface rlt0.0 {
    virtual-circuit-id 100;
    no-control-word;
  }
}
```

```
user@host# show protocols
mpls {
  no-cspf;
  interface all;
}
bgp {
  export local-routes;
  group internal {
    type internal;
    local-address 3.3.3.3;
    family inet {
      any;
    }
    family inet-vpn {
      unicast;
    }
    neighbor 4.4.4.4;
  }
}
ospf {
  area 0.0.0.0 {
```

```
      interface ge-5/3/8.0;
      interface ge-5/2/5.0;
      interface lo0.3 {
        passive;
      }
    }
  }
  ldp {
    interface all;
  }
  l2circuit {
    neighbor 2.2.2.2 {
      interface rlt0.0 {
        virtual-circuit-id 100;
        no-control-word;
      }
    }
  }
```

```
user@host# routing-instances
pe-vrf {
  instance-type vrf;
  interface rlt0.1;
  route-distinguisher 65056:1;
  vrf-import VPN-A-Import;
  vrf-export VPN-A-Export;
  vrf-table-label;
  protocols {
    ospf {
      export VPN-A-Import;
      area 0.0.0.0 {
        interface rlt0.1;
      }
    }
  }
}
```

```
user@host# policy-options
policy-statement VPN-A-Export {
  term a {
    then {
      community add VPN-A;
      accept;
    }
  }
  term b {
    then reject;
  }
}
policy-statement VPN-A-Import {
  term a {
    from {
      protocol bgp;
      community VPN-A;
      }
    then accept;
  }
```

```
    term b {
      then reject;
    }
}
policy-statement local-routes {
    then accept;
}
community VPN-A members target:100:100;
```

```
user@host# routing-options
router-id 3.3.3.3;
autonomous-system 65056;
```

Verification

Confirm that the configuration is working properly.

Verifying the Redundant Logical Tunnel Configuration

Purpose Verify that the redundant logical tunnel with the child logical tunnel interfaces are created with the correct encapsulations.

Action

```
user@host#  run show interfaces terse | match rlt0
lt-1/0/10.0             up    up   container--> rlt0.0
lt-1/0/10.1             up    up   container--> rlt0.1
lt-2/0/10.0             up    up   container--> rlt0.0
lt-2/0/10.1             up    up   container--> rlt0.1
rlt0                    up    up
rlt0.0                  up    up   ccc
rlt0.1                  up    up   inet     10.10.10.2/24
```

Verifying the Layer 2 Circuit

Purpose Verify that the Layer 2 circuit is up.

Action

```
user@host# run show l2circuit connections
Layer-2 Circuit Connections:

Legend for connection status (St)
EI -- encapsulation invalid      NP -- interface h/w not present
MM -- mtu mismatch               Dn -- down
EM -- encapsulation mismatch     VC-Dn -- Virtual circuit Down
CM -- control-word mismatch      Up -- operational
VM -- vlan id mismatch           CF -- Call admission control failure
OL -- no outgoing label          IB -- TDM incompatible bitrate
NC -- intf encaps not CCC/TCC    TM -- TDM misconfiguration
BK -- Backup Connection          ST -- Standby Connection
CB -- rcvd cell-bundle size bad  SP -- Static Pseudowire
LD -- local site signaled down   RS -- remote site standby
RD -- remote site signaled down  HS -- Hot-standby Connection
XX -- unknown

Legend for interface status
Up -- operational
Dn -- down
Neighbor: 2.2.2.2
    Interface                 Type  St     Time last up          # Up trans
    rlt0.0(vc 100)            rmt   Up     Aug  8 00:28:04 2013           1
      Remote PE: 2.2.2.2, Negotiated control-word: No
      Incoming label: 299776, Outgoing label: 299776
      Negotiated PW status TLV: No
      Local interface: rlt0.0, Status: Up, Encapsulation: VLAN
```

Verifying OSPF Neighbors

Purpose Verify that routers are adjacent and able to exchange OSPF data.

Action

```
user@host# run show ospf neighbor
Address          Interface              State     ID               Pri  Dead
30.30.30.2       ge-5/2/5.0             Full      4.4.4.4          128    38
20.20.20.1       ge-5/3/8.0             Full      2.2.2.2          128    38
```

Verifying the BGP Group

Purpose Verify that the BGP group is created.

Action

```
user@host# run show bgp group internal
Group Type: Internal    AS: 65056                 Local AS: 65056
    Name: internal        Index: 0                  Flags: <Export Eval>
    Export: [ local-routes ]
    Holdtime: 0
    Total peers: 1        Established: 1
    4.4.4.4+179
    inet.0: 1/6/3/0
    inet.2: 0/0/0/0
    bgp.l3vpn.0: 2/2/2/0
    pe-vrf.inet.0: 2/2/2/0
```

Verifying the BGP Routes in the Routing Table

Purpose Verify that the BGP routes are in the pe-vrf.inet.0 routing table.

Action

```
user@host# run show route protocol bgp table pe-vrf.inet.0
pe-vrf.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

50.50.50.0/24      *[BGP/170] 01:18:14, localpref 100, from 4.4.4.4
                      AS path: I, validation-state: unverified
                    > to 30.30.30.2 via ge-5/2/5.0, Push 16
50.50.51.0/24      *[BGP/170] 01:18:14, MED 2, localpref 100, from 4.4.4.4
                      AS path: I, validation-state: unverified
                    > to 30.30.30.2 via ge-5/2/5.0, Push 16
```

Related Documentation

- Configuring Redundant Logical Tunnels on page 30
- Redundant Logical Tunnels Overview on page 28

CHAPTER 6

Understanding Default PIM Tunnel Configurations

Configuring PIM Tunnels

PIM tunnels are enabled automatically on routers that have a tunnel PIC and on which you enable PIM sparse mode. You do not need to configure the tunnel interface.

PIM tunnels are unidirectional.

In PIM sparse mode, the first-hop router encapsulates packets destined for the rendezvous point (RP) router. The packets are encapsulated with a unicast header and are forwarded through a unicast tunnel to the RP. The RP then de-encapsulates the packets and transmits them through its multicast tree. To perform the encapsulation and de-encapsulation, the first-hop and RP routers must be equipped with Tunnel PICs.

The Junos OS creates two interfaces to handle PIM tunnels:

- **pe**—Encapsulates packets destined for the RP. This interface is present on the first-hop router.
- **pd**—De-encapsulates packets at the RP. This interface is present on the RP.

NOTE: **The pe and pd interfaces do not support class-of-service (CoS) configurations.**

Related Documentation

- Tunnel Services Overview on page 3

CHAPTER 7

Facilitating VRF Table Lookup Using Virtual Loopback Tunnel Interfaces

Configuring Virtual Loopback Tunnels for VRF Table Lookup

To enable egress filtering, you can either configure filtering based on the IP header, or you can configure a virtual loopback tunnel on routers equipped with a Tunnel PIC. Table 4 on page 43 describes each method.

Table 4: Methods for Configuring Egress Filtering

Method	Interface Type	Configuration Guidelines	Comments
Filter traffic based on the IP header	Nonchannelized Point-to-Point Protocol / High Level Data Link Control (PPP/HDLC) core-facing SONET/SDH interfaces	Include the **vrf-table-label** statement at the **[edit routing-instances *instance-name*]** hierarchy level. For more information, see the *Junos OS VPNs Library for Routing Devices*.	There is no restriction on customer-edge (CE) router-to-provider edge (PE) router interfaces.
Configure a virtual loopback tunnel on routers equipped with a Tunnel PIC	All interfaces	See the guidelines in this section.	Router must be equipped with a Tunnel PIC. There is no restriction on the type of core-facing interface used or CE router-to-PE router interface used. You cannot configure a virtual loopback tunnel and the **vrf-table-label** statement at the same time.

You can configure a virtual loopback tunnel to facilitate VRF table lookup based on MPLS labels. You might want to enable this functionality so you can do either of the following:

- Forward traffic on a PE router to CE device interface, in a shared medium, where the CE device is a Layer 2 switch without IP capabilities (for example, a metro Ethernet switch).

  The first lookup is done based on the VPN label to determine which VRF table to refer to, and the second lookup is done on the IP header to determine how to forward packets to the correct end hosts on the shared medium.

- Perform egress filtering at the egress PE router.

  The first lookup on the VPN label is done to determine which VRF table to refer to, and the second lookup is done on the IP header to determine how to filter and forward packets. You can enable this functionality by configuring output filters on the VRF interfaces.

To configure a virtual loopback tunnel to facilitate VRF table lookup based on MPLS labels, you specify a virtual loopback tunnel interface name and associate it with a routing instance that belongs to a particular routing table. The packet loops back through the virtual loopback tunnel for route lookup. To specify a virtual loopback tunnel interface name, you configure the virtual loopback tunnel interface at the **[edit interfaces]** hierarchy level and include the **family inet** and **family mpls** statements:

```
vt-fpc/pic/port {
    unit 0 {
        family inet;
        family mpls;
    }
    unit 1 {
        family inet;
    }
}
```

To associate the virtual loopback tunnel with a routing instance, include the virtual loopback tunnel interface name at the **[edit routing-instances]** hierarchy level:

```
interface vt-fpc/pic/port;
```

NOTE: On virtual loopback tunnel interfaces, none of the logical interface statements except the family statement is supported. Note that you can configure only inet and mpls families, and you cannot configure IPv4 or IPv6 addresses on virtual loopback tunnel interfaces. Also, virtual loopback tunnel interfaces do not support class-of-service (CoS) configurations.

Related Documentation

- Tunnel Services Overview on page 3
- Example: Configuring a Virtual Loopback Tunnel for VRF Table Lookup on page 45

Configuring Tunnel Interfaces for Routing Table Lookup

To configure tunnel interfaces to facilitate routing table lookups for VPNs, you specify a tunnel's endpoint IP addresses and associate them with a routing instance that belongs to a particular routing table. This enables the Junos OS to search in the appropriate routing table for the route prefix, because the same prefix can appear in multiple routing tables. To configure the destination VPN, include the **routing-instance** statement:

```
routing-instance {
  destination routing-instance-name;
}
```

You can include this statement at the following hierarchy levels:

- **[edit interfaces gr-*fpc/pic/port* unit *logical-unit-number* tunnel]**
- **[edit logical-systems *logical-system-name* interfaces gr-*fpc/pic/port* unit *logical-unit-number* tunnel]**

This configuration indicates that the tunnel's destination address is in routing instance ***routing-instance-name***. By default, the tunnel route prefixes are assumed to be in the default Internet routing table **inet.0**.

NOTE: If you configure a virtual loopback tunnel interface and the **vrf-table-label** statement on the same routing instance, the **vrf-table-label** statement takes precedence over the virtual loopback tunnel interface. For more information, see "Configuring Virtual Loopback Tunnels for VRF Table Lookup" on page 43.

For more information about VPNs, see the *Junos OS VPNs Library for Routing Devices*.

Related Documentation

- Tunnel Services Overview on page 3
- destination (Routing Instance) on page 74

Example: Configuring a Virtual Loopback Tunnel for VRF Table Lookup

Configure a virtual loopback tunnel for VRF table lookup:

```
[edit routing-instances]
routing-instance-1 {
  instance-type vrf;
  interface vt-1/0/0.0;
  interface so-0/2/2.0;
  route-distinguisher 2:3;
  vrf-import VPN-A-import;
  vrf-export VPN-A-export;
  routing-options {
    static {
      route 10.0.0.0/8 next-hop so-0/2/2.0;
    }
```

```
    }
  }
  routing-instance-2 {
    instance-type vrf;
    interface vt-1/0/0.1;
    interface so-0/3/2.0;
    route-distinguisher 4:5;
    vrf-import VPN-B-import;
    vrf-export VPN-B-export;
    routing-options {
      static {
        route 10.0.0.0/8 next-hop so-0/3/2.0;
      }
    }
  }
  [edit interfaces]
  vt-1/0/0 {
    unit 0 {
      family inet;
      family mpls;
    }
    unit 1 {
      family inet;
    }
  }
```

Related Documentation

- Tunnel Services Overview on page 3
- Configuring Virtual Loopback Tunnels for VRF Table Lookup on page 43

Example: Virtual Routing and Forwarding (VRF) and Service Configuration

The following example combines virtual routing and forwarding (VRF) and services configuration:

```
[edit policy-options]
policy-statement test-policy {
  term t1 {
    then reject;
  }
}
[edit routing-instances]
test {
  interface ge-0/2/0.0;
  interface sp-1/3/0.20;
  instance-type vrf;
  route-distinguisher 10.58.255.1:37;
  vrf-import test-policy;
  vrf-export test-policy;
  routing-options {
    static {
      route 0.0.0.0/0 next-table inet.0;
    }
  }
}
```

```
[edit interfaces]
ge-0/2/0 {
    unit 0 {
        family inet {
            service {
                input service-set nat-me;
                output service-set nat-me;
            }
        }
    }
}
sp-1/3/0 {
    unit 0 {
        family inet;
    }
    unit 20 {
        family inet;
        service-domain inside;
    }
    unit 21 {
        family inet;
        service-domain outside;
    }
    [edit services]
    stateful-firewall {
        rule allow-any-input {
            match-direction input;
            term t1 {
                then accept;
            }
        }
    }
    nat {
        pool hide-pool {
            address 10.58.16.100;
            port automatic;
        }
        rule hide-all-input {
            match-direction input;
            term t1 {
                then {
                    translated {
                        source-pool hide-pool;
                        translation-type source napt-44;
                    }
                }
            }
        }
    }
    service-set nat-me {
        stateful-firewall-rules allow-any-input;
        nat-rules hide-all-input;
        interface-service {
            service-interface sp-1/3/0.20;
        }
    }
```

}

CHAPTER 8

Enabling a VPN to Travel Through a Non-MPLS Network Using Dynamic Tunnels

Configuring Dynamic Tunnels

A VPN that travels through a non-MPLS network requires a GRE tunnel. This tunnel can be either a static tunnel or a dynamic tunnel. A static tunnel is configured manually between two PE routers. A dynamic tunnel is configured using BGP route resolution.

When a router receives a VPN route that resolves over a BGP next hop that does not have an MPLS path, a GRE tunnel can be created dynamically, allowing the VPN traffic to be forwarded to that route. Only GRE IPv4 tunnels are supported.

To configure a dynamic tunnel between two PE routers, include the **dynamic-tunnels** statement:

```
dynamic-tunnels tunnel-name {
  destination-networks prefix;
  source-address address;
}
```

You can configure this statement at the following hierarchy levels:

- **[edit routing-options]**
- **[edit routing-instances *routing-instance-name* routing-options]**
- **[edit logical-systems *logical-system-name* routing-options]**
- **[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options]**

For more information about configuring routing options or BGP, see the *Junos OS Routing Protocols Library for Routing Devices*. For more information about VPNs, see the *Junos OS VPNs Library for Routing Devices*.

Related Documentation

- Tunnel Services Overview on page 3

- dynamic-tunnels on page 77

PART 2

Encryption Services

CHAPTER 9

Overview

Encryption Overview

The IP Security (IPsec) architecture provides a security suite for the IP version 4 (IPv4) and IP version 6 (IPv6) network layers. The suite provides functionality such as authentication of origin, data integrity, confidentiality, replay protection, and nonrepudiation of source. It also defines mechanisms for key generation and exchange, management of security associations, and support for digital certificates.

IPsec defines a security association (SA) and key management framework that can be used with any network layer protocol. The SA specifies what protection policy to apply to traffic between two IP-layer entities. For more information, see the *Junos OS Administration Library for Routing Devices*. The standards are defined in the following RFCs:

- RFC 2401, *Security Architecture for the Internet Protocol*
- RFC 2406, *IP Encapsulating Security Payload (ESP)*

Related Documentation

- Configuring Encryption Interfaces on page 55
- Configuring Filters for Traffic Transiting the ES PIC on page 57
- Configuring ES PIC Redundancy on page 63
- Configuring IPsec Tunnel Redundancy on page 64

Configuring an ES Tunnel Interface for a Layer 3 VPN

To configure an ES tunnel interface for a Layer 3 VPN, you need to configure an ES tunnel interface on the provider edge (PE) router and on the customer edge (CE) router. You also need to configure IPsec on the PE and CE routers. For more information about configuring an ES tunnel for a Layer 3 VPN, see the *Junos OS VPNs Library for Routing Devices*.

Related Documentation

- Encryption Overview on page 53
- Configuring Encryption Interfaces on page 55
- Configuring Filters for Traffic Transiting the ES PIC on page 57
- Configuring ES PIC Redundancy on page 63
- Configuring IPsec Tunnel Redundancy on page 64

CHAPTER 10

Sending Encrypted Traffic Through Tunnels

Configuring Encryption Interfaces

When you configure the encryption interface, you associate the configured SA with a logical interface. This configuration defines the tunnel, including the logical unit, tunnel addresses, maximum transmission unit (MTU), optional interface addresses, and the name of the IPsec SA to apply to traffic. To configure an encryption interface, include the following statements at the **[edit interfaces es-*fpc/pic/port* unit *logical-unit-number*]** hierarchy level:

```
family inet {
    ipsec-sa ipsec-sa; # name of security association to apply to packet
    address address; # local interface address inside local VPN
    destination address; # destination address inside remote VPN
}
tunnel {
    source source-address;
    destination destination-address;
}
```

The addresses configured as the tunnel source and destination are the addresses in the outer IP header of the tunnel.

NOTE: You must configure the tunnel source address locally on the router, and the tunnel destination address must be a valid address for the security gateway terminating the tunnel.

The ES Physical Interface Card (PIC) is supported on M Series and T Series routers.

The SA must be a valid tunnel-mode SA. The interface address and destination address listed are optional. The destination address allows the user to configure a static route to

encrypt traffic. If a static route uses that destination address as the next hop, traffic is forwarded through the portion of the tunnel in which encryption occurs.

Specifying the Security Association Name for Encryption Interfaces

The security association is the set of properties that defines the protocols for encrypting Internet traffic. To configure encryption interfaces, you specify the SA name associated with the interface by including the **ipsec-sa** statement at the **[edit interfaces es-*fpc/pic/port* unit *logical-unit-number* family inet]** hierarchy level:

```
ipsec-sa sa-name;
```

For information about configuring the security association, see "Configuring Filters for Traffic Transiting the ES PIC" on page 57.

Configuring the MTU for Encryption Interfaces

The protocol MTU value for encryption interfaces must always be less than the default interface MTU value of 3900 bytes; the configuration fails to commit if you select a greater value. To set the MTU value, include the **mtu** statement at the **[edit interfaces *interface-name* unit *logical-unit-number* family inet]** hierarchy level:

```
mtu bytes;
```

For more information, see the *Junos OS Network Interfaces Library for Routing Devices*.

Example: Configuring an Encryption Interface

Configure an IPsec tunnel as a logical interface on the ES PIC. The logical interface specifies the tunnel through which the encrypted traffic travels. The **ipsec-sa** statement associates the security profile with the interface.

```
[edit interfaces]
es-0/0/0 {
  unit 0 {
    tunnel {
      source 10.5.5.5; # tunnel source address
      destination 10.6.6.6; # tunnel destination address
    }
    family inet {
      ipsec-sa manual-sa1; # name of security association to apply to packet
      mtu 3800;
      address 10.1.1.8/32 { # local interface address inside local VPN
      destination 10.2.2.254;  # destination address inside remote VPN
    }
  }
}
```

Related Documentation

- Encryption Overview on page 53
- Configuring Filters for Traffic Transiting the ES PIC on page 57
- Configuring ES PIC Redundancy on page 63
- Configuring IPsec Tunnel Redundancy on page 64

Configuring Filters for Traffic Transiting the ES PIC

This section contains the following topics:

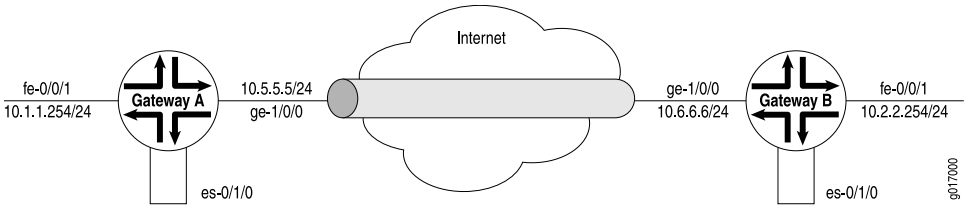
Traffic Overview

Traffic configuration defines the traffic that must flow through the tunnel. You configure outbound and inbound firewall filters, which identify and direct traffic to be encrypted and confirm that decrypted traffic parameters match those defined for the given tunnel. The outbound filter is applied to the LAN or WAN interface for the incoming traffic you want to encrypt. The inbound filter is applied to the ES PIC to check the policy for traffic coming in from the remote host. Because of the complexity of configuring a router to forward packets, no automatic checking is done to ensure that the configuration is correct.

NOTE: **The valid firewall filters statements for IPsec are destination-port, source-port, protocol, destination-address, and source-address.**

In Figure 3 on page 57, Gateway A protects the network **10.1.1.0/24**, and Gateway B protects the network **10.2.2.0/24**. The gateways are connected by an IPsec tunnel. For more information about firewalls, see the *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices*.

Figure 3: Example: IPsec Tunnel Connecting Security Gateways



The SA and ES interface for security Gateway A are configured as follows:

```
[edit security ipsec]
security-association manual-sa1 {
  manual {
    direction bidirectional {
      protocol esp;
      spi 2312;
      authentication {
        algorithm hmac-md5-96;
```

```
            key ascii-text 1234123412341234;
        }
        encryption {
            algorithm 3des-cbc;
            key ascii-text 12345678900987654321234;
        }
    }
  }
}
[edit interfaces es-0/1/0]
unit 0 {
  tunnel {
    source 10.5.5.5;
    destination 10.6.6.6;
  }
  family inet {
    ipsec-sa manual-sa1;
    address 10.1.1.8/32 {
      destination 10.2.2.254;
    }
  }
}
```

Configuring the Security Association

To configure the SA, include the **security-association** statement at the **[edit security]** hierarchy level:

```
security-association name {
  mode (tunnel | transport);
  manual {
    direction (inbound | outbound | bi-directional) {
      auxiliary-spi auxiliary-spi-value;
      spi spi-value;
      protocol (ah | esp | bundle);
      authentication {
        algorithm (hmac-md5-96 | hmac-sha1-96);
        key (ascii-text key | hexadecimal key);
      }
      encryption {
        algorithm (des-cbc | 3des-cbc);
        key (ascii-text key | hexadecimal key);
      }
    }
    dynamic {
      replay-window-size (32 | 64);
      ipsec-policy policy-name;
    }
  }
}
```

For more information about configuring an SA, see the *Junos OS Administration Library for Routing Devices*. For information about applying the SA to an interface, see "Specifying the Security Association Name for Encryption Interfaces" on page 56.

Configuring an Outbound Traffic Filter

To configure the outbound traffic filter, include the **filter** statement at the **[edit firewall]** hierarchy level:

```
filter filter-name {
   term term-name {
      from {
         match-conditions;
      }
      then {
         action;
         action-modifiers;
      }
   }
}
```

For more information, see the *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices*.

Example: Configuring an Outbound Traffic Filter

Firewall filters for outbound traffic direct the traffic through the desired IPsec tunnel and ensure that the tunneled traffic goes out the appropriate interface (see Figure 3 on page 57). Here, an outbound firewall filter is created on security Gateway A; it identifies the traffic to be encrypted and adds it to the input side of the interface that carries the internal virtual private network (VPN) traffic:

```
[edit firewall]
filter ipsec-encrypt-policy-filter {
   term term1 {
      from {
         source-address { # local network
         10.1.1.0/24;
      }
      destination-address { # remote network
      10.2.2.0/24;
   }
}
then ipsec-sa manual-sa1; # apply SA name to packet
term default {
   then accept;
}
```

NOTE: The source address, port, and protocol on the outbound traffic filter must match the destination address, port, and protocol on the inbound traffic filter. The destination address, port, and protocol on the outbound traffic filter must match the source address, port, and protocol on the inbound traffic filter.

Applying the Outbound Traffic Filter

After you have configured the outbound firewall filter, you apply it by including the **filter** statement at the **[edit interfaces *interface-name* unit *logical-unit-number* family inet]** hierarchy level:

```
filter {
  input filter-name;
}
```

Example: Applying the Outbound Traffic Filter

Apply the outbound traffic filter. The outbound filter is applied on the Fast Ethernet interface at the **[edit interfaces fe-0/0/1 unit 0 family inet]** hierarchy level. Any packet matching the IPsec action term (**term 1**) on the input filter (**ipsec-encrypt-policy-filter**), configured on the Fast Ethernet interface, is directed to the ES PIC interface at the **[edit interfaces es-0/1/0 unit 0 family inet]** hierarchy level. So, if a packet arrives from the source address **10.1.1.0/24** and goes to the destination address **10.2.2.0/24**, the Packet Forwarding Engine directs the packet to the ES PIC interface, which is configured with the **manual-sa1** SA. The ES PIC receives the packet, applies the **manual-sa1** SA, and sends the packet through the tunnel.

The router must have a route to the tunnel end point; add a static route if necessary.

```
[edit interfaces]
fe-0/0/1 {
  unit 0 {
    family inet {
      filter {
        input ipsec-encrypt-policy-filter;
      }
      address 10.1.1.254/24;
    }
  }
}
```

Configuring an Inbound Traffic Filter

To configure an inbound traffic filter, include the **filter** statement at the **[edit firewall]** hierarchy level:

```
filter filter-name {
  term term-name {
    from {
      match-conditions;
    }
    then {
      action;
      action-modifiers;
    }
  }
}
```

For more information, see the *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices*.

Example: Configuring an Inbound Traffic Filter

Configure an inbound firewall filter. This filter performs the final IPsec policy check and is created on security gateway A. The policy check ensures that only packets that match the traffic configured for this tunnel are accepted.

```
[edit firewall]
filter ipsec-decrypt-policy-filter {
   term term1 { # perform policy check
   from {
      source-address { # remote network
      10.2.2.0/24;
   }
   destination-address { # local network
   10.1.1.0/24;
}
then accept;
```

Applying the Inbound Traffic Filter to the Encryption Interface

After you create the inbound firewall filter, you can apply it to the ES PIC. To apply the filter to the ES PIC, include the **filter** statement at the **[edit interfaces es-*fpc/pic/port* unit *logical-unit-number* family inet filter]** hierarchy level:

```
filter {
   input filter;
}
```

The input filter is the name of the filter applied to received traffic. For a configuration example, see "Example: Configuring an Inbound Traffic Filter" on page 61. For more information about firewall filters, see the *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices*.

Example: Applying the Inbound Traffic Filter to the Encryption Interface

Apply the inbound firewall filter (**ipsec-decrypt-policy-filter**) to the decrypted packet to perform the final policy check. The IPsec **manual-sa1** SA is referenced at the **[edit interfaces es-1/2/0 unit 0 family inet]** hierarchy level and decrypts the incoming packet.

The Packet Forwarding Engine directs IPsec packets to the ES PIC. It uses the packet's security parameter index (SPI), protocol, and destination address to look up the SA configured on one of the ES interfaces. The IPsec **manual-sa1** SA is referenced at the **[edit interfaces es-1/2/0 unit 0 family inet]** hierarchy level and is used to decrypt the incoming packet. When the packets are processed (decrypted, authenticated, or both), the input firewall filter (**ipsec-decrypt-policy-filter**) is applied on the decrypted packet to perform the final policy check. **term1** defines the decrypted (and verified) traffic and performs the required policy check. For information about **term1**, see "Example: Configuring an Inbound Traffic Filter" on page 61.

NOTE: The inbound traffic filter is applied after the ES PIC has processed the packet, so the decrypted traffic is defined as any traffic that the remote gateway is encrypting and sending to this router. IKE uses this filter to determine the policy required for a tunnel. This policy is used during the negotiation with the remote gateway to find the matching SA configuration.

```
[edit interfaces]
es-1/2/0 {
  unit 0 {
    tunnel {
      source 10.5.5.5; # tunnel source address
      destination 10.6.6.6; # tunnel destination address
    }
    family inet {
      filter {
        input ipsec-decrypt-policy-filter;
      }
      ipsec-sa manual-sa1; # SA name applied to packet
      address 10.1.1.8/32 { # local interface address inside local VPN
      destination 10.2.2.254; # destination address inside remote VPN
    }
  }
}
```

Related Documentation

- Encryption Overview on page 53
- Configuring Encryption Interfaces on page 55
- Configuring ES PIC Redundancy on page 63
- Configuring IPsec Tunnel Redundancy on page 64

CHAPTER 11

Configuring Redundancy in Case of Service Failure

Configuring ES PIC Redundancy

You can configure ES PIC redundancy on M Series and T Series routers that have multiple ES PICs. With ES PIC redundancy, one ES PIC is active and another ES PIC is on standby. When the primary ES PIC has a servicing failure, the backup becomes active, inherits all the tunnels and SAs, and acts as the new next hop for IPsec traffic. Reestablishment of tunnels on the backup ES PIC does not require new Internet Key Exchange (IKE) negotiations. If the primary ES PIC comes online, it remains in standby and does not preempt the backup. To determine which PIC is currently active, use the **show ipsec redundancy** command.

NOTE: ES PIC redundancy is supported on M Series and T Series routers.

To configure an ES PIC as the backup, include the **backup-interface** statement at the **[edit interfaces *fpc/pic/port* es-options]** hierarchy level:

```
backup-interface es-fpc/pic/port;
```

Example: Configuring ES PIC Redundancy

After you create the inbound firewall filter, apply it to the master ES PIC. Here, the inbound firewall filter (**ipsec-decrypt-policy-filter**) is applied on the decrypted packet to perform the final policy check. The IPsec **manual-sa1** SA is referenced at the **[edit interfaces es-1/2/0 unit 0 family inet]** hierarchy level and decrypts the incoming packet. This example does not show SA and filter configuration. For information about SA and filter configuration, see the *Junos OS Administration Library for Routing Devices*, the *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices*, and "Example: Configuring an Inbound Traffic Filter" on page 61.

```
[edit interfaces]
es-1/2/0 {
   es-options {
```

```
        backup-interface es-1/0/0;
    }
    unit 0 {
        tunnel {
            source 10.5.5.5;
            destination 10.6.6.6;
        }
        family inet {
            ipsec-sa manual-sa1;
            filter {
                input ipsec-decrypt-policy-filter;
            }
            address 10.1.1.8/32 {
                destination 10.2.2.254;
            }
        }
    }
}
```

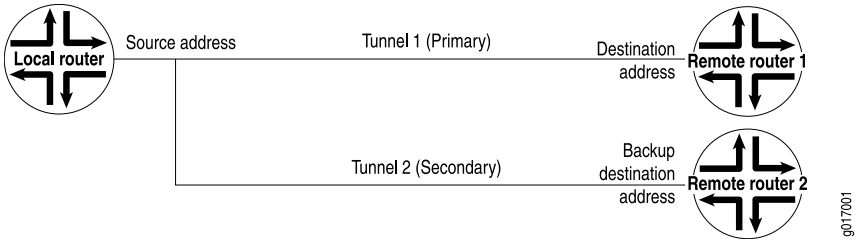
Related Documentation

- Encryption Overview on page 53
- Configuring Encryption Interfaces on page 55
- Configuring Filters for Traffic Transiting the ES PIC on page 57
- Configuring IPsec Tunnel Redundancy on page 64

Configuring IPsec Tunnel Redundancy

You can configure IPsec tunnel redundancy by specifying a backup destination address. The local router sends keepalives to determine the remote site's reachability. When the peer is no longer reachable, a new tunnel is established. For up to 60 seconds during failover, traffic is dropped without notification being sent. Figure 4 on page 64 shows IPsec primary and backup tunnels.

Figure 4: IPsec Tunnel Redundancy



To configure IPsec tunnel redundancy, include the **backup-destination** statement at the **[edit interfaces unit *logical-unit-number* tunnel]** hierarchy level:

```
backup-destinationaddress;
destination address;
source address;
```

NOTE: Tunnel redundancy is supported on M Series and T Series routers.

The primary and backup destinations must be on different routers.

The tunnels must be distinct from each other and policies must match.

For more information about tunnels, see *Tunnel Properties*.

Related Documentation

- Encryption Overview on page 53
- Configuring Encryption Interfaces on page 55
- Configuring Filters for Traffic Transiting the ES PIC on page 57
- Configuring ES PIC Redundancy on page 63

PART 3

Configuration Statements and Operational Commands

CHAPTER 12

Configuration Statements

address (Interfaces)

Syntax

```
address address {
  destination address;
}
```

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* family *family*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure the interface address.

Options ***address***—Address of the interface.

The remaining statement is explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Encryption Interfaces on page 55

allow-fragmentation

Syntax allow-fragmentation;

Hierarchy Level [edit interfaces gr-*fpc/pic/port* unit *logical-unit-number* tunnel],
[edit logical-systems *logical-system-name* interfaces gr-*fpc/pic/port* unit *logical-unit-number* tunnel]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable fragmentation of generic routing encapsulation (GRE) encapsulated packets regardless of maximum transmission unit (MTU) value.

Default By default, the GRE-encapsulated packets are dropped if the packet size exceeds the MTU setting of the egress interface.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- reassemble-packets on page 84
- Configuring Packet Reassembly on page 21

backup-destination

Syntax backup-destination *destination-address*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* tunnel],[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* tunnel]

Release Information Statement introduced before Junos OS Release 7.4.

Description For tunnel interfaces, specify the remote address of the backup tunnel.

Options ***destination-address***—Address of the remote side of the connection.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- destination (Interfaces) on page 73
- destination (Tunnel Remote End) on page 74
- Configuring IPsec Tunnel Redundancy on page 64

backup-interface

Syntax backup-interface *interface-name*;

Hierarchy Level [edit interfaces *interface-name* es-options]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure a backup ES Physical Interface Card (PIC). When the primary ES PIC has a servicing failure, the backup becomes active, inherits all the tunnels and security associations (SAs), and acts as the new next hop for IPsec traffic.

Options ***interface-name***—Name of ES interface to serve as the backup.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring ES PIC Redundancy on page 63

copy-tos-to-outer-ip-header

Syntax copy-tos-to-outer-ip-header;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced in Junos OS Release 8.2.

Description For GRE tunnel interfaces only, enable the inner IP header's ToS bits to be copied to the outer IP packet header.

Default If you omit this statement, the ToS bits in the outer IP header are set to 0.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring a GRE Tunnel to Copy ToS Bits to the Outer IP Header on page 21

destination (Interfaces)

Syntax destination *address*;

Hierarchy Level
[edit interfaces *interface-name* unit *logical-unit-number* tunnel]
[edit interfaces *interface-name* unit *logical-unit-number* family inet address *address*],
[edit interfaces *interface-name* unit *logical-unit-number* tunnel]
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family inet address *address*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For CoS on ATM interfaces, specify the remote address of the connection.

For point-to-point interfaces only, specify the address of the interface at the remote end of the connection.

For tunnel and encryption interfaces, specify the remote address of the tunnel.

Options ***address***—Address of the remote side of the connection.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Configuring Linear RED Profiles on ATM Interfaces*
- *Multilink and Link Services Logical Interface Configuration Overview*
- Configuring Encryption Interfaces on page 55
- *Configuring Traffic Sampling*
- *Configuring Flow Monitoring*
- Configuring Unicast Tunnels on page 17

destination (Routing Instance)

Syntax destination *routing-instance-name*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* tunnel routing-instance]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the destination routing instance that points to the routing table containing the tunnel destination address.

Default The default Internet routing table **inet.0**.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Tunnel Interfaces for Routing Table Lookup on page 45

destination (Tunnel Remote End)

Syntax destination *address*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* tunnel],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* tunnel]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1 for EX Series switches.

Description For tunnel interfaces, specify the remote address of the tunnel.

Options ***destination-address***—Address of the remote side of the connection.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Unicast Tunnels on page 17
- *Configuring Traffic Sampling*
- *Configuring Flow Monitoring*

destination-networks

Syntax destination-networks *prefix*;

Hierarchy Level
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*],
[edit logical-systems *logical-system-name* routing-options dynamic-tunnels *tunnel-name* rsvp-te *entry*],
[edit logical-systems *logical-system-name* routing-options dynamic-tunnels *tunnel-name*],
[edit routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*],
[edit routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name* rsvp-te *entry*],
[edit routing-options dynamic-tunnels *tunnel-name*],
[edit routing-options dynamic-tunnels *tunnel-name* rsvp-te *entry*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Description Specify the IPv4 prefix range for the destination network. Only tunnels within the specified IPv4 prefix range can be created.

Options ***prefix***—Destination prefix of the network.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation

- *Configuring GRE Tunnels for Layer 3 VPNs*
- Configuring Dynamic Tunnels on page 49
- *Configuring RSVP Automatic Mesh*

do-not-fragment

Syntax do-not-fragment;

Hierarchy Level [edit interfaces gr-*fpc/pic/port* unit *logical-unit-number* tunnel],
[edit logical-systems *logical-system-name* interfaces gr-*fpc/pic/port* unit *logical-unit-number* tunnel]

Release Information Statement introduced in Junos OS Release 9.2.

Description Set the do-not-fragment (DF) bit on the packets entering the GRE tunnel so that they do not get fragmented anywhere in the path.

Default By default, fragmentation is disabled.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- reassemble-packets on page 84
- Configuring Packet Reassembly on page 21

dynamic-tunnels

Syntax

```
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;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Description

Configure a dynamic tunnel between two PE routers.

Options

tunnel-name—Name of the dynamic tunnel.

The remaining statements are explained separately.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation

- *Example: Configuring a Two-Tiered Virtualized Data Center for Large Enterprise Networks*
- *Configuring GRE Tunnels for Layer 3 VPNs*
- Configuring Dynamic Tunnels on page 49

es-options

Syntax

```
es-options {
  backup-interface interface-name;
}
```

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description On ES interfaces, configure ES interface-specific interface properties.

The **backup-interface** statement is explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring ES PIC Redundancy on page 63

family

Syntax

```
family inet {
    ipsec-sa sa-name;
}
```

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure protocol family information for the logical interface.

Options *family*—Protocol family:

- **ccc**—Circuit cross-connect protocol suite
- **inet**—IP version 4 suite
- **inet6**—IP version 6 suite
- **iso**—Open Systems Interconnection (OSI) International Organization for Standardization (ISO) protocol suite
- **mlfr-end-to-end**—Multilink Frame Relay FRF.15
- **mlfr-uni-nni**—Multilink Frame Relay FRF.16
- **multilink-ppp**—Multilink Point-to-Point Protocol
- **mpls**—MPLS
- **tcc**—Translational cross-connect protocol suite
- **tnp**—Trivial Network Protocol
- **vpls**—Virtual private LAN service

The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Encryption Interfaces on page 55
- *Junos OS Network Interfaces Library for Routing Devices* for other statements that do not affect services interfaces.

filter

Syntax

```
filter {
  input filter-name;
  output filter-name;
}
```

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* family inet]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define the filters to be applied on an interface.

Options

input *filter-name*—Identifier for the input filter.

output *filter-name*—Identifier for the output filter.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Filters for Traffic Transiting the ES PIC on page 57

hold-time (OAM)

Syntax hold-time *seconds*;

Hierarchy Level

[edit protocols oam],
[edit protocols oam gre-tunnel interface *interface-name*]

Release Information Statement introduced in Junos OS Release 10.2.

Description Length of time the originating end of a GRE tunnel waits for keepalive packets from the other end of the tunnel before marking the tunnel as operationally down.

Options

seconds—Hold-time value.

Default: 5 seconds

Range: 5 through 250 seconds

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- GRE Keepalive Time Overview on page 9
- Configuring GRE Keepalive Time on page 9
- keepalive-time on page 82

interfaces

Syntax interfaces { ... }

Hierarchy Level [edit]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure interfaces on the router.

Default The management and internal Ethernet interfaces are automatically configured. You must configure all other interfaces.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- *Junos OS Network Interfaces Library for Routing Devices*

ipsec-sa

Syntax ipsec-sa *sa-name*;

Hierarchy Level [edit interfaces es-*fpc/pic/port* unit *logical-unit-number* family inet]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the IP Security (IPsec) SA name associated with the interface.

Options *sa-name*—IPsec SA name.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Encryption Interfaces on page 55
- *Junos OS Administration Library for Routing Devices*

keepalive-time

Syntax keepalive-time *seconds*;

Hierarchy Level [edit protocols oam],
[edit protocols oam gre-tunnel interface *interface-name*],
[edit protocols oam gre-tunnel interface *interface-name.unit-number*]

Release Information Statement introduced in Junos OS Release 10.2.

Description Time difference between consecutive keepalive packets in a GRE tunnel.

NOTE: Support for GRE keepalive packets on MPC line cards became available as of Junos OS Release 11.4.

Options *seconds*—Keepalive time value.

Default: 1 second

Range: 1 through 50 seconds

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- GRE Keepalive Time Overview on page 9
- Configuring GRE Keepalive Time on page 9
- hold-time on page 80

key

Syntax key *number*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* tunnel],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* tunnel]

Release Information Statement introduced before Junos OS Release 7.4.

Description Identify an individual traffic flow within a tunnel, as defined in RFC 2890, *Key and Sequence Number Extensions to GRE*. On M Series and T Series routers, you can configure the GRE interface on an Adaptive Services, Multiservices, or Tunnel PIC. On MX Series routers, configure the interface on a Multiservices DPC.

Options *number*—Value of the key.
Range: 0 through 4,294,967,295

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring a Key Number on GRE Tunnels on page 19

multicast-only

Syntax multicast-only;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* family inet],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family inet]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure the unit and family so that the interface can transmit and receive multicast traffic only. You can configure this property on the IP family only.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Restricting Tunnels to Multicast Traffic on page 23
- tunnel on page 92

peer-unit

Syntax peer-unit *unit-number*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure a peer relationship between two logical systems.

Options ***unit-number***—Peering logical system unit number.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Logical Tunnel Interfaces on page 25

reassemble-packets

Syntax reassemble-packets;

Hierarchy Level [edit interfaces gr-*fpc/pic/port* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces gr-*fpc/pic/port* unit *logical-unit-number*]

Release Information Statement introduced in Junos OS Release 9.2.

Description Enable reassembly of fragmented tunnel packets on generic routing encapsulation (GRE) tunnel interfaces.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Packet Reassembly on page 21

redundancy-group (Interfaces)

Syntax

```
redundancy-group {
  member-interface interface-name {
    (active | backup);
  }
}
```

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 13.3.

Description Configure member logical tunnels of redundant logical tunnels only on MX Series 3D Universal Edge Routers.

Options

active—Set the interface to the active mode.

backup—Set the interface to the backup mode.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To view this statement in the configuration.

Related Documentation

- Example: Configuring Redundant Logical Tunnels on page 31
- Configuring Redundant Logical Tunnels on page 30
- Redundant Logical Tunnels Overview on page 28
- **redundancy-group (Logical Tunnels) on page 86**

redundancy-group (Logical Tunnels)

Syntax

```
redundancy-group {
  interface-type {
    redundant-logical-tunnel {
      device count;
    }
  }
}
```

Hierarchy Level [edit chassis]

Release Information Statement introduced in Junos OS Release 13.3.
Support for up to 255 redundant logical tunnels added to Junos OS Release 13.3R3.

Description Configure redundant logical tunnels only on MX Series 3D Universal Edge Routers.

Options *count*—Specify the number of the redundant logical tunnels. For Junos OS Release 13.3R1, 13.3R2, and 14.1R1 the valid range is from 1 to 16. For Junos OS Release 13.3R3 the valid range is from 1 to 255.

—

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To view this statement in the configuration.

Related Documentation

- Example: Configuring Redundant Logical Tunnels on page 31
- Configuring Redundant Logical Tunnels on page 30
- Redundant Logical Tunnels Overview on page 28
- **redundancy-group (Interfaces) on page 85**

routing-instance

Syntax

```
routing-instance {
  destination routing-instance-name;
}
```

Hierarchy Level

[edit interfaces *interface-name* unit *logical-unit-number* tunnel],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* tunnel]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the destination routing instance that points to the routing table containing the tunnel destination address.

Default

The default Internet routing table **inet.0**.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Tunnel Interfaces for Routing Table Lookup on page 45

routing-instances

Syntax

routing-instances *routing-instance-name* { ... }

Hierarchy Level

[edit],
[edit logical-systems *logical-system-name*]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure an additional routing entity for a router. You can create multiple instances of BGP, IS-IS, OSPF, OSPF version 3 (OSPFv3), and RIP for a router.

Default

Routing instances are disabled for the router.

Options

routing-instance-name—Name of the routing instance, a maximum of 31 characters. The remaining statements are explained separately.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation

- *Configuring EVPN Routing Instances*
- *Configuring Routing Instances on PE Routers in VPNs*

routing-options

Syntax routing-options { ... }

Hierarchy Level [edit],
[edit logical-systems *logical-system-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name*],
[edit routing-instances *routing-instance-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure protocol-independent routing properties.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation
- *Protocol-Independent Routing Properties Feature Guide for Routing Devices*

source

Syntax source *source-address*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* family inet address *address*],
[edit interfaces *interface-name* unit *logical-unit-number* tunnel]

Release Information Statement introduced before Junos OS Release 7.4.

Description For tunnel and encryption interfaces, specify the source address.

Options ***source-address***—Address of the source side of the connection.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Encryption Interfaces on page 55
- *Configuring Traffic Sampling*
- *Configuring Flow Monitoring*

source

Syntax source *source-address*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* tunnel]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for the QFX Series.

Description Specify the source address of the tunnel.

Default If you do not specify a source address, the tunnel uses the unit's primary address as the source address of the tunnel.

Options ***source-address***—Address of the local side of the tunnel. This is the address that is placed in the outer IP header's source field.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- *Configuring Generic Routing Encapsulation Tunneling (CLI Procedure)*

source-address

Syntax source-address *address*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*],
[edit logical-systems *logical-system-name* routing-options dynamic-tunnels *tunnel-name*],
[edit routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*],
[edit routing-options dynamic-tunnels *tunnel-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Description Configure the tunnel source address.

Options ***address***—Name of the source address.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation
- Configuring Dynamic Tunnels on page 49

ttl

Syntax ttl *value*;

Hierarchy Level [edit interfaces *interface-name* unit *number* tunnel]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1 for EX Series switches.

Description Set the time-to-live value bit in the header of the outer IP packet.

Options ***value***—Time-to-live value.

Range: 0 through 255

Default: 64

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Tunnel Properties*
- *Configuring Generic Routing Encapsulation Tunneling (CLI Procedure)*

tunnel

Syntax

```
tunnel {
  backup-destination destination-address;
  destination destination-address;
  routing-instance {
    destination routing-instance-name;
  }
  source source-address;
  ttl number;
}
```

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure a tunnel. You can use the tunnel for unicast and multicast traffic or just for multicast traffic. You can also use tunnels for encrypted traffic or virtual private networks (VPNs).

The statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Encryption Interfaces on page 55
- *Tunnel Properties*
- *Junos OS VPNs Library for Routing Devices*

tunnel

Syntax

```
tunnel {
  allow-fragmentation;
  backup-destination address;
  destination destination-address;
  do-not-fragment;
  key number;
  routing-instance {
    destination routing-instance-name;
  }
  source source-address;
  ttl number;
}
```

Hierarchy Level

[edit interfaces *interface-name* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1 for EX Series switches.

Description

Configure a tunnel. You can use the tunnel for unicast and multicast traffic or just for multicast traffic. You can also use tunnels for encrypted traffic or virtual private networks (VPNs).

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Encryption Interfaces on page 55
- *Junos OS VPNs Library for Routing Devices*

unit (Interfaces)

Syntax

```
unit logical-unit-number {
  family inet {
    ipsec-sa sa-name;
  }
  tunnel {
    backup-destination destination-address;
    destination destination-address;
    routing-instance {
      destination routing-instance-name;
    }
    source source-address;
    ttl number;
  }
}
```

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.

Options ***logical-unit-number***—Number of the logical unit.

Range: 0 through 16,384

The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Encryption Interfaces on page 55
- *Junos OS Network Interfaces Library for Routing Devices*
- *Junos OS Network Interfaces Library for Routing Devices* for other statements that do not affect services interfaces.

unit (Interfaces)

Syntax

```
unit logical-unit-number {
  peer-unit unit-number;
  reassemble-packets;
  tunnel {
    allow-fragmentation;
    backup-destination address;
    destination destination-address;
    do-not-fragment;
    key number;
    routing-instance {
      destination routing-instance-name;
    }
    source source-address;
    ttl number;
  }
}
```

Hierarchy Level

[edit interfaces *interface-name*],
[edit logical-systems *logical-system-name* interfaces *interface-name*]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.

Options

logical-unit-number—Number of the logical unit.

Range: 0 through 16,384

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Junos OS Network Interfaces Library for Routing Devices* for other statements that do not affect services interfaces.

CHAPTER 13

Operational Commands

- clear ike security-associations
- clear ipsec security-associations
- request ipsec switch
- request security certificate (signed)
- request security certificate (unsigned)
- request security key-pair
- request system certificate add
- show ike security-associations
- show interfaces (Encryption)
- show interfaces (GRE)
- show interfaces (IP-over-IP)
- show interfaces (Logical Tunnel)
- show interfaces (Multicast Tunnel)
- show interfaces (PIM)
- show interfaces (Virtual Loopback Tunnel)
- show ipsec certificates
- show ipsec redundancy
- show ipsec security-associations
- show system certificate

clear ike security-associations

Syntax clear ike security-associations
<*destination-ip-address*>

Release Information Command introduced before Junos OS Release 7.4.

Description (Encryption interface on M Series and T Series routers only) Clear information about the current Internet Key Exchange (IKE) security association. This command is valid for dynamic security associations only. For IKEv2, this command creates new security associations for IKE SA and IPSEC SAs.

Options **none**—Clear all IKE security associations.

destination-ip-address—(Optional) Clear the IKE security association at the specified destination address.

Required Privilege Level view

Related Documentation
- show ike security-associations on page 105

List of Sample Output clear ike security-associations on page 96

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

Sample Output

clear ike security-associations

```
user@host> clear ike security-associations
```

clear ipsec security-associations

Syntax clear ipsec security-associations
<sa-name>

Release Information Command introduced before Junos OS Release 7.4.

Description (Encryption interface on M Series and T Series routers only) Clear information about the current IP Security (IPsec) security association. This command is valid for dynamic security associations only. For IKEv1, this command creates new security associations for IKE SA and IPSEC SAs.

Options **none**—Clear all IPsec security associations.

sa-name—(Optional) Clear the specified security association.

Required Privilege Level view

Related Documentation
- show ipsec security-associations on page 150

List of Sample Output clear ipsec security-associations on page 97

Output Fields See the show ipsec security-associations for an explanation of output fields.

Sample Output

clear ipsec security-associations

The following output from the **show ipsec security-associations detail** command is displayed before and after the **clear ipsec security-associations** command is issued:

```
user@host> show ipsec security-associations detail
Security association: sa-dynamic, Interface family: Up

  Direction: inbound, SPI: 242379418, State: Installed
  Mode: tunnel, Type: dynamic
  Protocol: ESP, Authentication: hmac-md5-96, Encryption: None
  Soft lifetime: Expires in 22979 seconds
  Hard lifetime: Expires in 28739 seconds

  Direction: outbound, SPI: 368592771, State: Installed
  Mode: tunnel, Type: dynamic
  Protocol: ESP, Authentication: hmac-md5-96, Encryption: None
  Soft lifetime: Expires in 22979 seconds
  Hard lifetime: Expires in 28739 seconds


user@host> clear ipsec security-associations


user@host> show ipsec security-associations detail
Security association: sa-dynamic, Interface family: Up

  Direction: inbound, SPI: 1031597683, State: Installed
```

```
Mode: tunnel, Type: dynamic
Protocol: ESP, Authentication: hmac-md5-96, Encryption: None
Soft lifetime: Expires in 23037 seconds
Hard lifetime: Expires in 28797 seconds

Direction: outbound, SPI: 1618419878, State: Installed
Mode: tunnel, Type: dynamic
Protocol: ESP, Authentication: hmac-md5-96, Encryption: None
Soft lifetime: Expires in 23037 seconds
Hard lifetime: Expires in 28797 seconds
```

request ipsec switch

Syntax request ipsec switch (interface <es-*fpc/pic/port*> | security-associations <*sa-name*>)

Release Information Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.

Description (Encryption interface on M Series, PTX Series, and T Series routers and EX Series switches only) Manually switch from the primary to the backup encryption services interface, or switch from the primary to the backup IP Security (IPsec) tunnel.

Options **interface <es-*fpc/pic/port*>**—Switch to the backup encryption interface.

security-associations <*sa-name*>—Switch to the backup tunnel.

Required Privilege Level view

Related Documentation
- show ipsec redundancy on page 148

List of Sample Output request ipsec switch on page 99

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

Sample Output

request ipsec switch

```
user@host> request ipsec switch security-associations sa-private
```

request security certificate (signed)

Syntax request security certificate enroll filename *filename* subject *subject* alternative-subject *alternative-subject* certification-authority *certification-authority* encoding (binary | pem) key-file *key-file* domain-name *domain-name*

Release Information Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.

Description (Encryption interface on M Series and T Series routers and EX Series switches only) Obtain a signed certificate from a certificate authority (CA). The signed certificate validates the CA and the owner of the certificate. The results are saved in a specified file to the **/var/etc/ikecert** directory.

Options

filename *filename*—File that stores the certificate.

subject *subject*—Distinguished name (**dn**), which consists of a set of components—for example, an organization (**o**), an organization unit (**ou**), a country (**c**), and a locality (**l**).

alternative-subject *alternative-subject*—Tunnel source address.

certification-authority *certification-authority*—Name of the certificate authority profile in the configuration.

encoding (binary | pem)—File format used for the certificate. The format can be a binary file or privacy-enhanced mail (PEM), an ASCII base64-encoded format. The default format is binary.

key-file *key-file*—File containing a local private key.

domain-name *domain-name*—Fully qualified domain name.

Required Privilege Level maintenance

List of Sample Output request security certificate (signed) on page 100

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

Sample Output

request security certificate (signed)

```
user@host> request security certificate enroll filename host.crt subject c=uk,o=london
alternative-subject 10.50.1.4 certification-authority verisign   key-file host-1.prv domain-name
host.juniper.net
CA name: juniper.net CA file: ca_verisign
local pub/private key pair: host.prv
subject: c=uk,o=london domain name: host.juniper.net
alternative subject: 10.50.1.4
Encoding: binary
Certificate enrollment has started. To view the status of your enrollment, check
 the key management process (kmd) log file at /var/log/kmd. <--------------
```


request security certificate (unsigned)

Syntax request security certificate enroll filename *filename* ca-file *ca-file* ca-name *ca-name* encoding (binary | perm) url *url*

Release Information Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.

Description (Encryption interface on M Series and T Series routers and EX Series switches only) Obtain a certificate from a certificate authority (CA). The results are saved in a specified file to the **/var/etc/ikecert** directory.

Options

filename *filename*—File that stores the public key certificate.

ca-file *ca-file*—Name of the certificate authority profile in the configuration.

ca-name *ca-name*—Name of the certificate authority.

encoding (binary | pem)—File format used for the certificate. The format can be a binary file or privacy-enhanced mail (PEM), an ASCII base64-encoded format. The default value is **binary**.

url *url*—Certificate authority URL.

Required Privilege Level maintenance

List of Sample Output request security certificate (unsigned) on page 102

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

Sample Output

request security certificate (unsigned)

```
user@host> request security certificate enroll filename ca_verisign ca-file verisign ca-name
juniper.net urlxyzcompany URL
http://<verisign ca-name xyzcompany url>/cgi-bin/pkiclient.exe CA name: juniper.net
 CA file: verisign Encoding: binary
Certificate enrollment has started. To view the status of your enrollment, check
 the key management process (kmd) log file at /var/log/kmd. <--------------
```

request security key-pair

Syntax request security key-pair *filename*
<size *key-size*>
<type (rsa | dsa)>

Release Information Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.

Description (Encryption interface on M Series and T Series routers and EX Series switches only) Generate a public and private key pair for a digital certificate.

Options ***filename***—Name of a file in which to store the key pair.

size *key-size*—(Optional) Key size, in bits. The key size can be **512**, **1024**, or **2048**. The default value is **1024**.

type—(Optional) Algorithm used to encrypt the key:

- **rsa**—RSA algorithm. This is the default.
- **dsa**—Digital signature algorithm with Secure Hash Algorithm (SHA).

Required Privilege Level maintenance

List of Sample Output request security key-pair on page 103

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

Sample Output

request security key-pair

```
user@host> request security key-pair security-key-file
```


request system certificate add

Syntax request system certificate add (*filename* | terminal)

Release Information Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description (Encryption interface on M Series and T Series routers, PTX Series, and QFX Series switches only) Add a certificate provided by the Juniper Networks certificate authority (CA).

Options ***filename***—Filename (URL, local, or remote).

terminal—Use login terminal.

Required Privilege Level maintenance

List of Sample Output request system certificate add on page 104

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

Sample Output

request system certificate add

```
user@host> request system certificate add terminal
```


show ike security-associations

Syntax
show ike security-associations
<brief | detail>
<*peer-address*>

Release Information
Command introduced before Junos OS Release 7.4.

Description
(Encryption interface on M Series and T Series routers only) Display information about Internet Key Exchange (IKE) security associations.

Options
none—Display standard information about all IKE security associations.

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

peer-address—(Optional) Display IKE security associations for the specified peer address.

Required Privilege Level
view

Related Documentation
- clear ike security-associations on page 96

List of Sample Output
show ike security-associations on page 108
show ike security-associations detail on page 108

Output Fields
Table 5 on page 105 lists the output fields for the **show ike security-associations** command. Output fields are listed in the approximate order in which they appear.

Table 5: show ike security-associations Output Fields

Field Name	Field Description	Level of Output
IKE peer	Remote end of the IKE negotiation.	**detail**
Role	Part played in the IKE session. The router triggering the IKE negotiation is the initiator, and the router accepting the first IKE exchange packets is the responder.	**detail**
Remote Address	Responder's address.	none specified
State	State of the IKE security association: • **Matured**—The IKE security association is established. • **Not matured**—The IKE security association is in the process of negotiation.	none specified
Initiator cookie	When the IKE negotiation is triggered, a random number is sent to the remote node.	All levels

Table 5: show ike security-associations Output Fields *(continued)*

Field Name	Field Description	Level of Output
Responder cookie	The remote node generates its own random number and sends it back to the initiator as a verification that the packets were received. Of the numerous security services available, protection against denial of service (DoS) is one of the most difficult to address. A “cookie” or anticlogging token (ACT) is aimed at protecting the computing resources from attack without spending excessive CPU resources to determine the cookie's authenticity. An exchange prior to CPU-intensive public key operations can thwart some DoS attempts (such as simple flooding with invalid IP source addresses).	All levels
Exchange type	Specifies the number of messages in an IKE exchange, and the payload types that are contained in each message. Each exchange type provides a particular set of security services, such as anonymity of the participants, perfect forward secrecy of the keying material, and authentication of the participants. Junos OS supports two types of exchanges: • **Main**—The exchange is done with six messages. **Main** encrypts the payload, protecting the identity of the neighbor. • **Aggressive**—The exchange is done with three messages. **Aggressive** does not encrypt the payload, leaving the identity of the neighbor unprotected.	All Levels
Authentication method	Type of authentication determines which payloads are exchanged and when they are exchanged. The Junos OS supports only **pre-shared keys**.	**detail**
Local	Prefix and port number of the local end.	**detail**
Remote	Prefix and port number of the remote end.	**detail**
Lifetime	Number of seconds remaining until the IKE security association expires.	**detail**
Algorithms	Header for the IKE algorithms output. • **Authentication**—Type of authentication algorithm used:**md5** or **sha1**. • **Encryption**—Type of encryption algorithm used: **des-cbc**, **3des-cbc**, or **None**. • **Pseudo random function**—Function that generates highly unpredictable random numbers:**hmac-md5** or**hmac-sha1**.	**detail**
Traffic statistics	Number of bytes and packets received and transmitted on the IKE security association. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the IKE security association. • **Input packets, Output packets**—Number of packets received and transmitted on the IKE security association.	**detail**

Table 5: show ike security-associations Output Fields *(continued)*

Field Name	Field Description	Level of Output
Flags	Notification to the key management process of the status of the IKE negotiation: • **caller notification sent**—Caller program notified about the completion of the IKE negotiation. • **waiting for done**—Negotiation is done. The library is waiting for the remote end retransmission timers to expire. • **waiting for remove**—Negotiation has failed. The library is waiting for the remote end retransmission timers to expire before removing this negotiation. • **waiting for policy manager**—Negotiation is waiting for a response from the policy manager.	**detail**
IPsec security associates	Number of IPsec security associations created and deleted with this IKE security association.	**detail**

Phase 2 negotiations in progress	Number of phase 2 IKE negotiations in progress and status information:
	• **Negotiation type**—Type of phase 2 negotiation. The Junos OS currently supports **quick mode**.
	• **Message ID**—Unique identifier for a phase 2 negotiation.
	• **Local identity**—Identity of the local phase 2 negotiation. The format is *id-type-name* (*proto-name*:*port-number*,[0..*id-data-len*] = *iddata-presentation*)
	• **Remote identity**—Identity of the remote phase 2 negotiation. The format is *id-type-name* (*proto-name*:*port-number*,[0..*id-data-len*] = *iddata-presentation*)
	• **Flags**—Notification to the key management process of the status of the IKE negotiation:
	  ◦ **caller notification sent**—Caller program notified about the completion of the IKE negotiation.
	  ◦ **waiting for done**—Negotiation is done. The library is waiting for the remote end retransmission timers to expire.
	  ◦ **waiting for remove**—Negotiation has failed. The library is waiting for the remote end retransmission timers to expire before removing this negotiation.
	  ◦ **waiting for policy manager**—Negotiation is waiting for a response from the policy manager.
		detail

Sample Output

show ike security-associations

```
user@host> show ike security-associations
Remote Address  State         Initiator cookie  Responder cookie  Exchange type
4.4.4.4         Matured       93870456fa000011  723a20713700003e  Main
```

show ike security-associations detail

```
user@host>  show ike security-associations detail
IKE peer 4.4.4.4
  Role: Initiator, State: Matured
  Initiator cookie: cf22bd81a7000001, Responder cookie: fe83795c2800002e
  Exchange type: Main, Authentication method: Pre-shared-keys
  Local: 4.4.4.5:500, Remote: 4.4.4.4:500
  Lifetime: Expires in 187 seconds
  Algorithms:
   Authentication        : md5
   Encryption            : 3des-cbc
   Pseudo random function: hmac-md5
  Traffic statistics:
   Input  bytes  :                 1000
   Output bytes  :                 1280
   Input  packets:                    5
   Output packets:                    9
  Flags: Caller notification sent
  IPsec security associations: 2 created, 0 deleted
  Phase 2 negotiations in progress: 1

Negotiation type: Quick mode, Role: Initiator, Message ID: 3582889153
    Local: 4.4.4.5:500, Remote: 4.4.4.4:500
    Local identity: ipv4_subnet(tcp:80,[0..7]=10.1.1.0/24)
    Remote identity: ipv4_subnet(tcp:100,[0..7]=10.1.2.0/24)
    Flags: Caller notification sent, Waiting for done
```

show interfaces (Encryption)

Syntax

show interfaces es-*fpc/pic/port:channel*
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index *snmp-index*>
<statistics>

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M Series and T Series routers only) Display status information about the specified encryption interface.

Options

es-*fpc/pic/port:channel*—Display standard status information about the specified encryption interface.

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

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

List of Sample Output

show interfaces (Encryption) on page 112
show interfaces brief (Encryption) on page 112
show interfaces detail (Encryption) on page 112
show interfaces extensive (Encryption) on page 113

Output Fields

Table 6 on page 109 lists the output fields for the **show interfaces** (ES) command. Output fields are listed in the approximate order in which they appear.

Table 6: Encryption show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.	All levels
Interface index	Physical interface's index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none

Table 6: Encryption show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Type	Encapsulation being used on the interface.	All levels
Link-level type	Encapsulation being used on the physical interface.	All levels
MTU	MTU size on the physical interface.	All levels
Speed	Speed at which the interface is running.	All levels
Device flags	Information about the physical device. Possible values are described in the "Link Flags" section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Input rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output rate	Output rate in bps and pps.	None specified
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface. • **Input packets, Output packets**—Number of packets received and transmitted on the interface. • **Anti-replay failures**—Total number of antireplay failures seen on all tunnels configured on the ES PIC. • **Authentication**—Total number of authentication failures seen on all tunnels configured on the ES PIC.	**detail extensive**
Egress queues	Total number of egress queues supported on the specified interface.	**detail extensive**
Queue counters	CoS queue number and its associated user-configured forwarding class name. • **Queued packets**—Number of queued packets. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism.	**detail extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number, which reflects its initialization sequence.	**detail extensive** none

Table 6: Encryption show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Flags	Information about the logical interface. Possible values are described in the "Logical Interface Flags" section under *Common Output Fields Description*.	All levels
IP-Header	IP header of the logical interface.	All levels
Encapsulation	Encapsulation on the logical interface.	All levels
protocol-family	Protocol family configured on the logical interface. If the protocol is **inet**, the IP address of the interface is also displayed.	**brief**
Input packets	Number of packets received on the logical interface.	None specified
Output packets	Number of packets transmitted on the logical interface.	None specified
Traffic statistics	Total number of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Local statistics	Statistics for traffic received from and transmitted to the Routing Engine. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Transit statistics	Statistics for traffic transiting the router. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Protocol	Protocol family configured on the logical interface, such as **iso**, **inet6**, **mpls**.	**detail extensive** none
MTU	MTU size on the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route table	Routing table in which the logical interface address is located. For example, **0** refers to the routing table inet.0.	**detail extensive**
Flags	Information about the protocol family flags. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**detail extensive** none
Addresses, Flags	Information about the address flags. Possible values are described in the "Addresses Flags" section under *Common Output Fields Description*.Address	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none

Table 6: Encryption show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Local	IP address of the logical interface.	**detail extensive** none
Broadcast	Broadcast address of the logical interface.	**detail extensive**
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

Sample Output

show interfaces (Encryption)

```
user@host> show interfaces es-0/3/0
Physical interface: es-0/3/0, Enabled, Physical link is Up
  Interface index: 138, SNMP ifIndex: 71
  Type: IPSEC, Link-level type: IPSEC-over-IP, MTU: 3900, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface es-0/3/0.0 (Index 70) (SNMP ifIndex 45)
    Flags: Hardware-Down Point-To-Point SNMP-Traps
    IP-Header 10.0.10.2:10.0.10.1::df:64:00000000 Encapsulation: IPSEC
    Input packets : 0
    Output packets: 0
    Protocol inet, MTU: 3800
      Flags: None
      Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
        Destination: 10.10.0.2, Local: 10.10.0.1
```

show interfaces brief (Encryption)

```
user@host> show interfaces es-0/3/0 brief
Physical interface: es-0/3/0, Enabled, Physical link is Up
  Type: IPSEC, Link-level type: IPSEC-over-IP, MTU: 3900, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps

  Logical interface es-0/3/0.0
    Flags: Hardware-Down Point-To-Point SNMP-Traps
    IP-Header 10.0.10.2:10.0.10.1::df:64:00000000 Encapsulation: IPSEC
    inet  10.10.0.1        --> 10.10.0.2s
```

show interfaces detail (Encryption)

```
user@host> show interfaces es-0/3/0 detail
Physical interface: es-0/3/0, Enabled, Physical link is Up
  Interface index: 138, SNMP ifIndex: 71, Generation: 21
  Type: IPSEC, Link-level type: IPSEC-over-IP, MTU: 3900, Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
```

```
 Input  packets:                    0                    0 pps
 Output packets:                    0                    0 pps
 Anti-replay failures       : 0
 Authentication failures    : 0
Egress queues: 4 supported, 4 in use
Queue counters:       Queued packets  Transmitted packets      Dropped packets

  0 best-effort                    0                    0                    0

  1 expedited-fo                   0                    0                    0

  2 assured-forw                   0                    0                    0

  3 network-cont                   0                    0                    0


Logical interface es-0/3/0.0 (Index 70) (SNMP ifIndex 45) (Generation 9)
  Flags: Hardware-Down Point-To-Point SNMP-Traps
  IP-Header 10.0.10.2:10.0.10.1::df:64:00000000 Encapsulation: IPSEC
  Traffic statistics:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
  Local statistics:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
  Transit statistics:
   Input  bytes  :                   0                    0 bps
   Output bytes  :                   0                    0 bps
   Input  packets:                   0                    0 pps
   Output packets:                   0                    0 pps
  Protocol inet, MTU: 3800, Generation: 22, Route table: 0
    Flags: None
    Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
      Destination: 10.10.0.2, Local: 10.10.0.1, Broadcast: Unspecified,
      Generation: 26
```

show interfaces extensive (Encryption)

```
user@host> show interfaces es-0/3/0 extensive
Physical interface: es-0/3/0, Enabled, Physical link is Up
  Interface index: 138, SNMP ifIndex: 71, Generation: 21
  Type: IPSEC, Link-level type: IPSEC-over-IP, MTU: 3900, Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                    0 bps
   Output bytes  :                   0                    0 bps
   Input  packets:                   0                    0 pps
   Output packets:                   0                    0 pps
   Anti-replay failures       : 0
   Authentication failures    : 0
  Egress queues: 4 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                    0                    0                    0
```

```
  1 expedited-fo                 0                 0                 0

  2 assured-forw                 0                 0                 0

  3 network-cont                 0                 0                 0


Logical interface es-0/3/0.0 (Index 70) (SNMP ifIndex 45) (Generation 9)
  Flags: Hardware-Down Point-To-Point SNMP-Traps
  IP-Header 10.0.10.2:10.0.10.1::df:64:00000000 Encapsulation: IPSEC
  Traffic statistics:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
  Local statistics:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
  Transit statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps
  Protocol inet, MTU: 3800, Generation: 22, Route table: 0
    Flags: None
    Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
      Destination: 10.10.0.2, Local: 10.10.0.1, Broadcast: Unspecified,
      Generation: 26
```

show interfaces (GRE)

Syntax

show interfaces *interface-type*
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index *snmp-index*>
<statistics>

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for the QFX Series.

Description

Display status information about the specified generic routing encapsulation (GRE) interface.

Options

interface-type—On M Series and T Series routers and EX Series switches, the interface type is **gr-*fpc/pic/port***.

brief | detail | extensive | terse—(Optional) Display the specified output level of interface information.

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

NOTE: You can configure generic routing encapsulation (GRE) interfaces (gre-x/y/z) only for GMPLS control channels. GRE interfaces are not supported or configurable for other applications. For more information about GMPLS, see the *Junos OS MPLS Applications Library for Routing Devices* and the *Junos OS, Release 14.1*.

Required Privilege Level

view

List of Sample Output

show interfaces (GRE) on page 119
show interfaces brief (GRE) on page 119
show interfaces detail (GRE) on page 119
show interfaces detail (GRE) on an EX4200 Virtual Chassis Member Switch on page 120
show interfaces extensive (GRE) on page 121

Output Fields

Table 7 on page 116 lists the output fields for the **show interfaces** (GRE) command. Output fields are listed in the approximate order in which they appear.

Table 7: GRE show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.	All levels
Interface index	Physical interface's index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Type	Type of interface.	All levels
Link-level type	Encapsulation used on the physical interface.	All levels
MTU	MTU size on the physical interface.	All levels
Speed	Speed at which the interface is running.	All levels
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Device Flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	All levels
Interface Flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Input rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output rate	Output rate in bps and pps.	None specified
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	The number of and the rate at which input and output bytes and packets are received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number, which reflects its initialization sequence.	**detail extensive** none

Table 7: GRE show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support.	**detail extensive**
Flags	Information about the logical interface. Possible values listed in the "Logical Interface Flags" section under *Common Output Fields Description*. describe general information about the logical interface. GRE-specific information about the logical interface is indicated by the presence or absence of the following value in this field: • **Reassemble-Pkts**—If the **Flags** field includes this string, the GRE tunnel is configured to reassemble tunnel packets that were fragmented after tunnel encapsulation.	All levels
IP-Header	IP header of the logical interface. If the **tunnel key** statement is configured, this information is included in the **IP Header** entry. GRE-specific information about the logical interface is indicated by the presence or absence of the following value in this field: • **df**—If the **IP-Header** field includes this string immediately following the 16 bits of identification information (that is, if **:df:** displays after the twelfth byte), the GRE tunnel is configured to allow fragmentation of GRE packets after encapsulation.	All levels
Encapsulation	Encapsulation on the logical interface.	All levels
Copy-tos-to-outer-ip-header	Status of type of service (ToS) bits in the GRE packet header: • **On**—ToS bits were copied from the payload packet header into the header of the IP packet sent through the GRE tunnel. • **Off**—ToS bits were not copied from the payload packet header and are set to 0 in the GRE packet header. NOTE: EX Series switches do not support copying ToS bits to the encapsulated packet, so the value of this field is always **Off** in switch output.	**detail extensive**
Gre keepalives configured	Indicates whether a GRE keepalive time and hold time are configured for the GRE tunnel. NOTE: EX Series switches do not support configuration of GRE tunnel keepalive times and hold times, so the value of this field is always **Off** in switch output.	**detail extensive**
Gre keepalives adjacency state	Status of the other end of the GRE tunnel: **Up** or **Down**. If keepalive messages are not received by either end of the GRE tunnel within the hold-time period, the GRE keepalive adjacency state is down even when the GRE tunnel is up.	**detail extensive**
Input packets	Number of packets received on the logical interface.	None specified
Output packets	Number of packets transmitted on the logical interface.	None specified

Table 7: GRE show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Traffic statistics	Rate of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize. • **Input rate**—Rate of bits and packets received on the interface. • **Output rate**—Rate of bits and packets transmitted on the interface.	**detail extensive**
Local statistics	Statistics for traffic received from and transmitted to the Routing Engine. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Transit statistics	Statistics for traffic transiting the router. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive** none
Protocol	Protocol family configured on the logical interface, such as **iso**, **inet6**, or **mpls**.	**detail extensive none**
protocol-family	Protocol family configured on the logical interface. If the protocol is **inet**, the IP address of the interface is also displayed.	**brief**
MTU	MTU size on the logical interface.	**detail extensive none**
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route table	Routing table in which the logical interface address is located. For example, **0** refers to the routing table inet.0.	**detail extensive**
Flags	Information about the protocol family flags. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**detail extensive** none
Addresses, Flags	Information about the address flags. Possible values are described in the "Addresses Flags" section under *Common Output Fields Description*.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none
Local	IP address of the logical interface.	**detail extensive** none
Broadcast	Broadcast address of the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

Sample Output

show interfaces (GRE)

```
user@host> show interfaces gr-1/2/0
Physical interface: gr-0/0/0, Enabled, Physical link is Up
  Interface index: 132, SNMP ifIndex: 26
  Type: GRE, Link-level type: GRE, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface gr-0/0/0.0 (Index 68) (SNMP ifIndex 47)
    Flags: Point-To-Point SNMP-Traps 16384
    IP-Header 1.1.1.2:1.1.1.1:47:df:64:0000000000000000 Encapsulation: GRE-NULL
  Input packets : 0
  Output packets: 0
    Protocol inet, MTU: 1476
      Flags: None
      Addresses, Flags: Is-Primary
        Local: 1.10.1.1
```

show interfaces brief (GRE)

```
user@host> show interfaces gr-1/2/0 brief
Physical interface: gr-1/2/0, Enabled, Physical link is Up
  Type: GRE, Link-level type: GRE, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps

  Logical interface gr-1/2/0.0
    Flags: Hardware-Down Point-To-Point SNMP-Traps 0x4000
    IP-Header 10.10.0.2:10.10.0.1:47:df:64:0000000000000000
    Encapsulation: GRE-NULL
    inet  10.100.0.1/30
    mpls
```

show interfaces detail (GRE)

```
user@host> show interfaces gr-1/2/0 detail
Physical interface: gr-0/0/0, Enabled, Physical link is Up
  Interface index: 132, SNMP ifIndex: 26, Generation: 13
  Type: GRE, Link-level type: GRE, MTU: Unlimited, Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps

  Logical interface gr-0/0/0.0 (Index 68) (SNMP ifIndex 47) (Generation 8)
    Flags: Point-To-Point SNMP-Traps 16384
    IP-Header 1.1.1.2:1.1.1.1:47:df:64:0000000000000000 Encapsulation: GRE-NULL
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
```

```
    Output packets:                    0
  Local statistics:
    Input  bytes  :                    0
    Output bytes  :                    0
    Input  packets:                    0
    Output packets:                    0
  Transit statistics:
    Input  bytes  :                    0                    0 bps
    Output bytes  :                    0                    0 bps
    Input  packets:                    0                    0 pps
    Output packets:                    0                    0 pps
  Protocol inet, MTU: 1476, Generation: 12, Route table: 0
    Flags: None
    Addresses, Flags: Is-Primary
      Destination: Unspecified, Local: 1.10.1.1, Broadcast: Unspecified,
      Generation: 15
```

show interfaces detail (GRE) on an EX4200 Virtual Chassis Member Switch

```
user@switch> show interfaces gr-2/0/15 detail
Physical interface: gr-2/0/15, Enabled, Physical link is Up
  Interface index: 195, SNMP ifIndex: 846, Generation: 198
  Type: GRE, Link-level type: GRE, MTU: Unlimited, Speed: 1000mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:1f:12:38:0f:d2, Hardware address: 00:1f:12:38:0f:d2
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Statistics last cleared: 2011-09-14 17:43:15 UTC (00:00:18 ago)
  Traffic statistics:
   Input  bytes  :              5600636                    0 bps
   Output bytes  :              5600636                    0 bps
   Input  packets:                20007                    0 pps
   Output packets:                20007                    0 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0

  Logical interface gr-2/0/15.0 (Index 75) (SNMP ifIndex 847) (HW Token 4093)
  (Generation 140)
    Flags: Point-To-Point SNMP-Traps 0x0
    IP-Header 180.20.30.2:180.20.20.3:47:df:64:0000000000000000
    Encapsulation: GRE-NULL
    Copy-tos-to-outer-ip-header: Off
    Gre keepalives configured: Off, Gre keepalives adjacency state: down
    Traffic statistics:
     Input  bytes  :              5600886
     Output bytes  :              2881784
     Input  packets:                20010
     Output packets:                10018
    Local statistics:
     Input  bytes  :                  398
     Output bytes  :                  264
     Input  packets:                    5
     Output packets:                    3
    Transit statistics:
     Input  bytes  :              5600488                    0 bps
     Output bytes  :              2881520                    0 bps
     Input  packets:                20005                    0 pps
     Output packets:                10015                    0 pps
```

```
  Protocol inet, Generation: 159, Route table: 0
    Flags: None
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 90.90.90/24, Local: 90.90.90.10, Broadcast: 90.90.90.255,
      Generation: 144

Logical interface gr-2/0/15.1 (Index 80) (SNMP ifIndex 848) (HW Token 4088)
(Generation 150)
  Flags: Point-To-Point SNMP-Traps 0x0
  IP-Header 160.20.40.2:160.20.30.1:47:df:64:0000000000000000
  Encapsulation: GRE-NULL
  Copy-tos-to-outer-ip-header: Off
  Gre keepalives configured: Off, Gre keepalives adjacency state: down
  Traffic statistics:
   Input  bytes  :                  260
   Output bytes  :              2880148
   Input  packets:                    4
   Output packets:                10002
  Local statistics:
   Input  bytes  :                  112
   Output bytes  :                    0
   Input  packets:                    2
   Output packets:                    0
  Transit statistics:
   Input  bytes  :                  148                    0 bps
   Output bytes  :              2880148                    0 bps
   Input  packets:                    2                    0 pps
   Output packets:                10002                    0 pps
  Protocol inet, Generation: 171, Route table: 0
    Flags: None
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 70.70.70/24, Local: 70.70.70.10, Broadcast: 70.70.70.255,
      Generation: 160
```

show interfaces extensive (GRE)

The output for the **show interfaces extensive** command is identical to that for the **show interfaces detail** command. For sample output, see show interfaces detail (GRE) on page 119 and show interfaces detail (GRE) on an EX4200 Virtual Chassis Member Switch on page 120.

show interfaces (IP-over-IP)

Syntax
show interfaces *interface-type*
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index *snmp-index*>
<statistics>

Release Information
Command introduced before Junos OS Release 7.4.

Description
Display status information about the specified IP-over-IP interface.

Options
interface-type—On M Series and T Series routers, the interface type is **ip-*fpc/pic/port***.

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

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

Required Privilege Level
view

List of Sample Output
show interfaces (IP-over-IP) on page 124
show interfaces brief (IP-over-IP) on page 125
show interfaces detail (IP-over-iP) on page 125
show interfaces extensive (IP-over-IP) on page 125

Output Fields
Table 8 on page 122 lists the output fields for the **show interfaces** (IP-over-IP) command. Output fields are listed in the approximate order in which they appear.

Table 8: IP-over-IP show interfaces Output Fields

Field	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.	All levels
Interface index	Physical interface's index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

Table 8: IP-over-IP show interfaces Output Fields *(continued)*

Field	Field Description	Level of Output
Type	Type of interface.	All levels
Link-level type	Encapsulation used on the physical interface.	All levels
MTU	MTU size on the physical interface.	All levels
Speed	Speed at which the interface is running.	All levels
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Device flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Input rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output rate	Output rate in bps and pps.	None specified
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support.	**detail extensive**
Flags	Information about the logical interface. Possible values are described in the "Logical Interface Flags" section under *Common Output Fields Description*.	All levels
IP Header	IP header of the logical interface.	All levels
Encapsulation	Encapsulation on the logical interface.	All levels
Input packets	Number of packets received on the logical interface.	None specified

Table 8: IP-over-IP show interfaces Output Fields *(continued)*

Field	Field Description	Level of Output
Output packets	Number of packets transmitted on the logical interface.	None specified
Traffic statistics	Total number of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize. • **Input rate**—Rate of bits and packets received on the interface. • **Output rate**—Rate of bits and packets transmitted on the interface.	**detail extensive**
Local statistics	Statistics for traffic received from and transmitted to the Routing Engine. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Transit statistics	Statistics for traffic transiting the router. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Protocol	Protocol family configured on the logical interface, such as **iso**, **inet6**, or **mpls**.	**detail extensive none**
protocol-family	Protocol family configured on the logical interface. If the protocol is **inet**, the IP address of the interface is also displayed.	**brief**
MTU	MTU size on the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route table	Routing table in which the logical interface address is located. For example, **0** refers to the routing table inet.0.	**detail extensive**
Flags	Information about the protocol family flags. Possible values are described in the “Family Flags” section under *Common Output Fields Description*.	**detail extensive** none

Sample Output

show interfaces (IP-over-IP)

```
user@host> show interfaces ip-0/0/0
Physical interface: ip-0/0/0, Enabled, Physical link is Up
  Interface index: 133, SNMP ifIndex: 27
  Type: IPIP, Link-level type: IP-over-IP, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface ip-0/0/0.0 (Index 69) (SNMP ifIndex 49)
    Flags: Point-To-Point SNMP-Traps 16384
    IP-Header 2.2.2.1:2.2.2.2:4:df:64:00000000 Encapsulation: IPv4-NULL
  Input packets : 0
```

```
  Output packets: 0
    Protocol inet, MTU: 1480
      Flags: None
```

show interfaces brief (IP-over-IP)

```
user@host> show interfaces ip-0/0/0 brief
Physical interface: ip-0/0/0, Enabled, Physical link is Up
  Type: IPIP, Link-level type: IP-over-IP, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: SNMP-Traps

  Logical interface ip-0/0/0.0
    Flags: Point-To-Point SNMP-Traps 16384
    IP-Header 2.2.2.1:2.2.2.2:4:df:64:00000000 Encapsulation: IPv4-NULL
    inet
```

show interfaces detail (IP-over-iP)

```
user@host> show interfaces ip-0/0/0 detail
Physical interface: ip-0/0/0, Enabled, Physical link is Up
  Interface index: 133, SNMP ifIndex: 27, Generation: 14
  Type: IPIP, Link-level type: IP-over-IP, MTU: Unlimited, Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps

  Logical interface ip-0/0/0.0 (Index 69) (SNMP ifIndex 49) (Generation 9)
    Flags: Point-To-Point SNMP-Traps 16384
    IP-Header 2.2.2.1:2.2.2.2:4:df:64:00000000 Encapsulation: IPv4-NULL
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Local statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Transit statistics:
     Input  bytes  :                   0                   0 bps
     Output bytes  :                   0                   0 bps
     Input  packets:                   0                   0 pps
     Output packets:                   0                   0 pps
    Protocol inet, MTU: 1480, Generation: 13, Route table: 0
      Flags: None
```

show interfaces extensive (IP-over-IP)

The output for the show interfaces extensive command is identical to that for the show interfaces detail command. For sample output, see show interfaces detail (IP-over-iP) on page 125.

show interfaces (Logical Tunnel)

Syntax show interfaces *interface-type*
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index *snmp-index*>
<statistics>

Release Information Command introduced before Junos OS Release 7.4.

Description Display status information about the specified logical tunnel interface.

Options ***interface-type***—On M Series and T Series routers, the interface type is **lt-*fpc/pic/port***.

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

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

Required Privilege Level view

List of Sample Output show interfaces extensive (Logical Tunnel) on page 130

Output Fields Table 9 on page 126 lists the output fields for the **show interfaces** (logical tunnel) command. Output fields are listed in the approximate order in which they appear.

Table 9: Logical Tunnel show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.	All levels
Interface index	Physical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

Table 9: Logical Tunnel show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Type	Type of interface. **Software-Pseudo** indicates a standard software interface with no associated hardware device.	All levels
Link-level type	Encapsulation used on the physical interface.	All levels
MTU	MTU size on the physical interface.	All levels
Clocking	Reference clock source: **Internal or External** when configured. Otherwise, **Unspecified**.	All levels
Speed	Speed at which the interface is running.	All levels
Device flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Link type	Type of link.	All levels
Link flags	Information about the link. Possible values are described in the "Link Flags" section under *Common Output Fields Description*.	All levels
Physical info	Information about the physical interface.	All levels
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Current address	Configured MAC address.	**detail extensive** none
Hardware address	Hardware MAC address.	**detail extensive** none
Alternate link address	Backup link address.	**detail extensive** none
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped:** ***year-month-day hour:minute:second timezone*** **(*****hour:minute:second*** **ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago)**.	**detail extensive** none
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface. • **Input packets, Output packets**—Number of packets received and transmitted on the interface.	**detail extensive**

Table 9: Logical Tunnel show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Input errors	Input errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious: • **Errors**—Sum of the incoming frame aborts and FCS errors. • **Drops**—Number of packets dropped by the input queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **Framing errors**—Number of packets received with an invalid frame checksum (FCS). • **Runts**—Number of frames received that are smaller than the runt threshold. • **Giants**—Number of frames received that are larger than the giant threshold. • **Policed discards**—Number of frames that the incoming packet match code discarded because they were not recognized or not of interest. Usually, this field reports protocols that the Junos OS does not handle. • **Resource errors**—Sum of transmit drops.	**extensive**
Output errors	Output errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious: • **Carrier transitions**—Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If the number of carrier transitions increments quickly (perhaps once every 10 seconds), the cable, the far-end system, or the PIC is malfunctioning. • **Errors**—Sum of the outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **MTU errors**—Number of packets larger than the MTU threshold. • **Resource errors**—Sum of transmit drops.	**extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP interface index number.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Flags	Information about the logical interface. Possible values are described in the "Logical Interface Flags" section under *Common Output Fields Description*.	All levels
Encapsulation	Encapsulation on the logical interface.	All levels

Table 9: Logical Tunnel show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Traffic statistics	Total number of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize. • **Input bytes**—Rate of bytes received on the interface. • **Output bytes**—Rate of bytes transmitted on the interface. • **Input packets**—Rate of packets received on the interface. • **Output packets**—Rate of packets transmitted on the interface.	**detail extensive**
Local statistics	Statistics for traffic received from and transmitted to the Routing Engine. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Transit statistics	Statistics for traffic transiting the router. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Protocol	Protocol family configured on the logical interface, such as **iso**, **inet6**, **mpls**.	**detail extensive** none
MTU	MTU size on the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route table	Route table in which this address exists. For example, **Route table:0** refers to inet.0.	**detail extensive**
Flags	Information about the protocol family flags. Possible values are described in the “Family Flags” section under *Common Output Fields Description*.	**detail extensive** none
Addresses, Flags	Information about the address flags. Possible values are described in the “Addresses Flags” section under *Common Output Fields Description*.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none
Local	IP address of the logical interface.	**detail extensive** none
Broadcast	Broadcast address of the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

Sample Output

show interfaces extensive (Logical Tunnel)

```
user@host> show interfaces lt-1/0/0 extensive
Physical interface: lt-1/0/0, Enabled, Physical link is Up
  Interface index: 143, SNMP ifIndex: 70, Generation: 26
  Type: Logical-tunnel, Link-level type: Logical-tunnel, MTU: 0,
  Clocking: Unspecified, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Unspecified
  Link flags     : None
  Physical info  : 13
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:90:69:a6:48:7e, Hardware address: Unspecified
  Alternate link address: Unspecified
  Last flapped   : 2004-03-03 15:53:52 PST (22:08:46 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0,
    Policed discards: 0
  Output errors:
    Carrier transitions: 1, Errors: 0, Drops: 0, MTU errors: 0

  Logical interface lt-1/0/0.0 (Index 66) (SNMP ifIndex 467) (Generation 3024)
    Flags: Point-To-Point SNMP-Traps 16384 DLCI 100 Encapsulation: FR-NLPID
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Local statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Transit statistics:
     Input  bytes  :                   0                   0 bps
     Output bytes  :                   0                   0 bps
     Input  packets:                   0                   0 pps
     Output packets:                   0                   0 pps
    Protocol inet, MTU: 4470, Generation: 7034, Route table: 0
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.1.1/24, Local: 10.1.1.1, Broadcast: Unspecified,
        Generation: 2054
```

show interfaces (Multicast Tunnel)

Syntax

show interfaces *interface-type*
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index *snmp-index*>
<statistics>

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display status information about the specified multicast tunnel interface and its logical encapsulation and de-encapsulation interfaces.

Options

interface-type—On M Series and T Series routers, the interface type is **mt-*fpc/pic/port***.

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

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

Additional Information

The multicast tunnel interface has two logical interfaces: encapsulation and de-encapsulation. These interfaces are automatically created by the Junos OS for every multicast-enabled VPN routing and forwarding (VRF) instance. The encapsulation interface carries multicast traffic traveling from the edge interface to the core interface. The de-encapsulation interface carries traffic coming from the core interface to the edge interface.

Required Privilege Level

view

List of Sample Output

Output Fields

Table 10 on page 132 lists the output fields for the **show interfaces** (Multicast Tunnel) command. Output fields are listed in the approximate order in which they appear.

Table 10: Multicast Tunnel show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.	All levels
Interface index	Physical interface's index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Type	Type of interface.	All levels
Link-level type	Encapsulation used on the physical interface.	All levels
MTU	MTU size on the physical interface.	All levels
Speed	Speed at which the interface is running.	All levels
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Device flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Input Rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output Rate	Output rate in bps and pps.	None specified
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**

Table 10: Multicast Tunnel show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	All levels

Sample Output

show interfaces (Multicast Tunnel)

```
user@host> show interfaces mt-1/2/0
Physical interface: mt-1/2/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 41
  Type: Multicast-GRE, Link-level type: GRE, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
```

show interfaces brief (Multicast Tunnel)

```
user@host> show interfaces mt-1/2/0 brief
Physical interface: mt-1/2/0, Enabled, Physical link is Up
  Type: Multicast-GRE, Link-level type: GRE, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: SNMP-Traps
```

show interfaces detail (Multicast Tunnel)

```
user@host> show interfaces mt-1/2/0 detail
Physical interface: mt-1/2/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 41, Generation: 28
  Type: Multicast-GRE, Link-level type: GRE, MTU: Unlimited, Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Statistics last cleared: Never
   Traffic statistics:
    Input  bytes  :           170664562              560000 bps
    Output bytes  :           112345376              368176 bps
    Input  packets:             2439107                1000 pps
    Output packets:             2439120                1000 pps
```

show interfaces extensive (Multicast Tunnel)

```
user@host> show interfaces mt-1/2/0 extensive
Physical interface: mt-1/2/0, Enabled, Physical link is Up
   Interface index: 141, SNMP ifIndex: 529, Generation: 144
   Type: Multicast-GRE, Link-level type: GRE, MTU: Unlimited, Speed: 800mbps
   Hold-times     : Up 0 ms, Down 0 ms
   Device flags   : Present Running
   Interface flags: SNMP-Traps
   Statistics last cleared: Never
```

```
   Traffic statistics:
    Input  bytes  :            170664562                560000 bps
    Output bytes  :            112345376                368176 bps
    Input  packets:              2439107                  1000 pps
    Output packets:              2439120                  1000 pps
    IPv6 transit statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0

   Logical interface mt-1/2/0.32768 (Index 83) (SNMP ifIndex 556) (Generation
148)
     Flags: Point-To-Point SNMP-Traps 0x4000 IP-Header
232.1.1.1:10.0.0.6:47:df:64:0000000800000000 Encapsulation: GRE-NULL
     Traffic statistics:
      Input  bytes  :            170418430
      Output bytes  :            112070294
      Input  packets:              2434549
      Output packets:              2435593
      IPv6 transit statistics:
       Input  bytes  :                    0
       Output bytes  :                    0
       Input  packets:                    0
       Output packets:                    0
     Local statistics:
      Input  bytes  :                    0
      Output bytes  :                80442
      Input  packets:                    0
      Output packets:                 1031
     Transit statistics:
      Input  bytes  :            170418430                560000 bps
      Output bytes  :            111989852                368176 bps
      Input  packets:              2434549                  1000 pps
      Output packets:              2434562                  1000 pps
      IPv6 transit statistics:
       Input  bytes  :                    0
       Output bytes  :                    0
       Input  packets:                    0
       Output packets:                    0
     Protocol inet, MTU: 1572, Generation: 182, Route table: 4
       Flags: None
     Protocol inet6, MTU: 1572, Generation: 183, Route table: 4
       Flags: None

   Logical interface mt-1/2/0.1081344 (Index 84) (SNMP ifIndex 560) (Generation
149)
     Flags: Point-To-Point SNMP-Traps 0x6000 Encapsulation: GRE-NULL
     Traffic statistics:
      Input  bytes  :               246132
      Output bytes  :               355524
      Input  packets:                 4558
      Output packets:                 4558
      IPv6 transit statistics:
       Input  bytes  :                    0
       Output bytes  :                    0
       Input  packets:                    0
       Output packets:                    0
     Local statistics:
      Input  bytes  :               246132
      Output bytes  :                    0
```

```
    Input  packets:                 4558
    Output packets:                    0
   Transit statistics:
    Input  bytes  :                    0                    0 bps
    Output bytes  :               355524                    0 bps
    Input  packets:                    0                    0 pps
    Output packets:                 4558                    0 pps
    IPv6 transit statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
   Protocol inet, MTU: Unlimited, Generation: 184, Route table: 4
     Flags: None
   Protocol inet6, MTU: Unlimited, Generation: 185, Route table: 4
     Flags: None
```

show interfaces (Multicast Tunnel Encapsulation)

```
user@host> show interfaces mt-3/1/0.32768
Logical interface mt-3/1/0.32768 (Index 67) (SNMP ifIndex 0)
    Flags: Point-To-Point SNMP-Traps 0x4000
    IP-Header 239.1.1.1:10.255.70.15:47:df:64:0000000800000000
    Encapsulation: GRE-NULL
  Input packets : 0
  Output packets: 2
    Protocol inet, MTU: Unlimited
      Flags: None
```

show interfaces (Multicast Tunnel De-Encapsulation)

```
user@host> show interfaces mt-3/1/0.49152
Logical interface mt-3/1/0.49152 (Index 74) (SNMP ifIndex 0)
    Flags: Point-To-Point SNMP-Traps 0x6000 Encapsulation: GRE-NULL
  Input packets : 0
  Output packets: 2
    Protocol inet, MTU: Unlimited
      Flags: None
```

show interfaces (PIM)

Syntax

show interfaces *interface-type*
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index *snmp-index*>
<statistics>

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display status information about the specified Protocol Independent Multicast (PIM) de-encapsulation or PIM encapsulation interface, respectively.

Options

interface-type—On M Series and T Series routers, the PIM de-encapsulation interface type is **pd-*fpc/pic/port*** and the PIM encapsulation interface type is **pe-*fpc/pic/port***.

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

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

List of Sample Output

show interfaces (PIM De-Encapsulation) on page 137
show interfaces brief (PIM De-Encapsulation) on page 138
show interfaces detail (PIM De-Encapsulation) on page 138
show interfaces extensive (PIM Encapsulation) on page 138
show interfaces (PIM Encapsulation) on page 138
show interfaces brief (PIM Encapsulation) on page 138
show interfaces detail (PIM Encapsulation) on page 139
show interfaces extensive (PIM Encapsulation) on page 139

Output Fields

Table 11 on page 136 lists the output fields for the **show interfaces** (PIM de-encapsulation or encapsulation) command. Output fields are listed in the approximate order in which they appear.

Table 11: PIM show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels

Table 11: PIM show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.	All levels
Interface index	Physical interface's index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Type	Type of interface.	All levels
Link-level type	Encapsulation used on the physical interface.	All levels
MTU	MTU size on the physical interface.	All levels
Speed	Speed at which the interface is running.	All levels
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Device flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Input Rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output Rate	Output rate in bps and pps.	None specified
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**

Sample Output

show interfaces (PIM De-Encapsulation)

```
user@host> show interfaces pd-0/0/0
Physical interface: pd-0/0/0, Enabled, Physical link is Up
  Interface index: 130, SNMP ifIndex: 25
  Type: PIMD, Link-level type: PIM-Decapsulator, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
```

```
  Interface flags: SNMP-Traps
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
```

show interfaces brief (PIM De-Encapsulation)

```
user@host> show interfaces pd-0/0/0 brief
Physical interface: pd-0/0/0, Enabled, Physical link is Up
  Type: PIMD, Link-level type: PIM-Decapsulator, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: SNMP-Traps
```

show interfaces detail (PIM De-Encapsulation)

```
user@host> show interfaces pd-0/0/0 detail
Physical interface: pd-0/0/0, Enabled, Physical link is Up
  Interface index: 130, SNMP ifIndex: 25, Generation: 11
  Type: PIMD, Link-level type: PIM-Decapsulator, MTU: Unlimited, Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps
```

show interfaces extensive (PIM Encapsulation)

```
user@host> show interfaces pd-0/0/0 extensive
Physical interface: pd-0/0/0, Enabled, Physical link is Up
  Interface index: 130, SNMP ifIndex: 25, Generation: 11
  Type: PIMD, Link-level type: PIM-Decapsulator, MTU: Unlimited, Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps
```

show interfaces (PIM Encapsulation)

```
user@host> show interfaces pe-0/0/0
Physical interface: pe-0/0/0, Enabled, Physical link is Up
  Interface index: 131, SNMP ifIndex: 26
  Type: PIME, Link-level type: PIM-Encapsulator, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
```

show interfaces brief (PIM Encapsulation)

```
user@host> show interfaces pe-0/0/0 brief
Physical interface: pe-0/0/0, Enabled, Physical link is Up
  Type: PIME, Link-level type: PIM-Encapsulator, MTU: Unlimited, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: SNMP-Traps
```

show interfaces detail (PIM Encapsulation)

```
user@host> show interfaces pe-0/0/0 detail
Physical interface: pe-0/0/0, Enabled, Physical link is Up
  Interface index: 131, SNMP ifIndex: 26, Generation: 12
  Type: PIME, Link-level type: PIM-Encapsulator, MTU: Unlimited, Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                    0                    0 bps
   Output bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
   Output packets:                    0                    0 pps
```

show interfaces extensive (PIM Encapsulation)

```
user@host> show interfaces pe-0/0/0 extensive
Physical interface: pe-0/0/0, Enabled, Physical link is Up
  Interface index: 131, SNMP ifIndex: 26, Generation: 12
  Type: PIME, Link-level type: PIM-Encapsulator, MTU: Unlimited, Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Interface flags: SNMP-Traps
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                    0                    0 bps
   Output bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
   Output packets:                    0                    0 pps
```

show interfaces (Virtual Loopback Tunnel)

Syntax

show interfaces vt-*fpc/pic/port*
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index *snmp-index*>
<statistics>

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display status information about the specified virtual loopback tunnel interface.

Options

vt-*fpc/pic/port*—Display standard information about the specified virtual loopback tunnel interface.

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

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

List of Sample Output

show interfaces (Virtual Loopback Tunnel) on page 142
show interfaces brief (Virtual Loopback Tunnel) on page 143
show interfaces detail (Virtual Loopback Tunnel) on page 143
show interfaces extensive (Virtual Loopback Tunnel) on page 143

Output Fields

Table 12 on page 140 lists the output fields for the **show interfaces** (virtual loopback tunnel) command. Output fields are listed in the approximate order in which they appear.

Table 12: Virtual Loopback Tunnel show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels
Enabled	State of the interface. Possible values are described in the “Enabled Field” section under *Common Output Fields Description*.	All levels
Interface index	Physical interface's index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none

Table 12: Virtual Loopback Tunnel show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Type	Type of interface.	All levels
Link-level type	Encapsulation used on the physical interface.	All levels
MTU	MTU size on the physical interface.	All levels
Speed	Speed at which the interface is running.	All levels
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.	**detail extensive**
Device flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	All levels
Input Rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output Rate	Output rate in bps and pps.	None specified
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface. • **Input packets, Output packets**—Number of packets received and transmitted on the interface.	**detail extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Flags	Information about the logical interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Encapsulation	Encapsulation on the logical interface.	All levels
Input packets	Number of packets received on the logical interface.	None specified
Output packets	Number of packets transmitted on the logical interface.	None specified

Table 12: Virtual Loopback Tunnel show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Bandwidth	Bandwidth allotted to the logical interface, in kilobytes per second.	All levels
Traffic statistics	Total number of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.	**detail extensive**
Transit statistics	Statistics for traffic transiting the router. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
protocol-family	Protocol family configured on the logical interface. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**brief**
Protocol	Protocol family configured on the logical interface. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**detail extensive** none
MTU	Maximum transmission unit size on the logical interface.	**detail extensive** none
Maximum labels	Maximum number of MPLS labels configured for the MPLS protocol family on the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route Table	Routing table in which the logical interface address is located. For example, **0** refers to the routing table inet.0.	**detail extensive**
Flags	Information about protocol family flags. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**detail extensive** none

Sample Output

show interfaces (Virtual Loopback Tunnel)

```
user@host> show interfaces vt-1/2/0
Physical interface: vt-1/2/0, Enabled, Physical link is Up
  Interface index: 144, SNMP ifIndex: 40
  Type: Loopback, Link-level type: Virtual-loopback-tunnel, MTU: Unlimited,
  Speed: 800mbps
  Device flags   : Present Running
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface vt-1/2/0.0 (Index 76) (SNMP ifIndex 57)
    Flags: Point-To-Point 16384 Encapsulation: Virtual-loopback-tunnel
```

```
  Input packets : 0
  Output packets: 0
    Protocol inet, MTU: Unlimited
      Flags: None
    Protocol mpls, MTU: Unlimited, Maximum labels: 3
      Flags: None
```

show interfaces brief (Virtual Loopback Tunnel)

```
user@host> show interfaces vt-1/2/0 brief
Physical interface: vt-1/2/0, Enabled, Physical link is Up
  Type: Loopback, Link-level type: Virtual-loopback-tunnel, MTU: Unlimited,
  Speed: 800mbps
  Device flags   : Present Running

  Logical interface vt-1/2/0.0
    Flags: Point-To-Point 16384 Encapsulation: Virtual-loopback-tunnel
    inet
    mpls
```

show interfaces detail (Virtual Loopback Tunnel)

```
user@host> show interfaces vt-1/2/0 detail
Physical interface: vt-1/2/0, Enabled, Physical link is Up
  Interface index: 144, SNMP ifIndex: 40, Generation: 27
  Type: Loopback, Link-level type: Virtual-loopback-tunnel, MTU: Unlimited,
  Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
  Device flags   : Present Running
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps

  Logical interface vt-1/2/0.0 (Index 76) (SNMP ifIndex 57) (Generation 17)
    Flags: Point-To-Point 16384 Encapsulation: Virtual-loopback-tunnel
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Transit statistics:
     Input  bytes  :                   0                   0 bps
     Output bytes  :                   0                   0 bps
     Input  packets:                   0                   0 pps
     Output packets:                   0                   0 pps
    Protocol inet, MTU: Unlimited, Generation: 33, Route table: 0
      Flags: None
   Protocol mpls, MTU: Unlimited, Maximum labels: 3, Generation: 34, Route table:
 0
      Flags: None
```

show interfaces extensive (Virtual Loopback Tunnel)

```
user@host> show interfaces vt-1/2/0 extensive
Physical interface: vt-1/2/0, Enabled, Physical link is Up
  Interface index: 144, SNMP ifIndex: 40, Generation: 27
  Type: Loopback, Link-level type: Virtual-loopback-tunnel, MTU: Unlimited,
  Speed: 800mbps
  Hold-times     : Up 0 ms, Down 0 ms
```

```
 Device flags   : Present Running
 Statistics last cleared: Never
 Traffic statistics:
  Input  bytes  :                   0                   0 bps
  Output bytes  :                   0                   0 bps
  Input  packets:                   0                   0 pps
  Output packets:                   0                   0 pps

 Logical interface vt-1/2/0.0 (Index 76) (SNMP ifIndex 57) (Generation 17)
   Flags: Point-To-Point 16384 Encapsulation: Virtual-loopback-tunnel
   Traffic statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
   Transit statistics:
    Input  bytes  :                   0                   0 bps
    Output bytes  :                   0                   0 bps
    Input  packets:                   0                   0 pps
    Output packets:                   0                   0 pps
   Protocol inet, MTU: Unlimited, Generation: 33, Route table: 0
     Flags: None
  Protocol mpls, MTU: Unlimited, Maximum labels: 3, Generation: 34, Route table:
0
     Flags: None
```

show ipsec certificates

Syntax show ipsec certificates
<brief | detail>
<crl *crl-name* | *serial-number*>

Release Information Command introduced before Junos OS Release 7.4.

Description (Encryption interface on M Series and T Series routers only) Display information about the IPsec certificate database.

Options **none**—Display standard information about all of the entries in the IPsec certificate database.

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

crl *crl-name* | *serial-number*—(Optional) Display information about the entries on the certificate revocation list (CRL) or for the specified serial number. A CRL is a timestamped list identifying revoked certificates. The CRL is signed by a certificate authority (CA) or CRL issuer and made freely available in a public repository. Each revoked certificate is identified in a CRL by its certificate serial number.

Required Privilege Level view

Related Documentation
- clear ipsec security-associations on page 97

List of Sample Output show ipsec certificates detail on page 146

Output Fields Table 13 on page 145 lists the output fields for the **show ipsec certificates** command. Output fields are listed in the approximate order in which they appear.

Table 13: show ipsec certificates Output Fields

Field Name	Field Description	Level of Output
Database	Display information about the IPsec certificate database. • **Total entries**—Number of database entries, including entries that are not trusted or that are in the process of being deleted. • **Active entries**—Number of database entries, excluding entries that are marked as deleted. • **Locked entries**—Number of statically configured database entries that cannot expire, such as CA certificates that are root or trusted.	All levels
Subject	Distinguished name for the certificate for **C**, **O**, **CN**, as described in RFC 3280, *Internet x.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile.*	All levels
ID	Identification number of the database entry. **ID** is generated by the internal certificate database.	All levels

Table 13: show ipsec certificates Output Fields *(continued)*

Field Name	Field Description	Level of Output
References	Reference number the certificate manager has for the particular entry.	**detail**
Serial	Unique serial number assigned to each certificate by the CA.	All levels
Flags	State of the certificate. • **Trusted**—Passed validity checks. • **Not trusted**—Failed validity checks. • **Root**—Entry is locked and may have been learned through IKE or a locally configured CA certificate. • **Non-root**—Entry is not locked. • **Crl-issuer**—Entity issues CRLs. • **Non-crl-issuer**—Entity does not issue CRLs.	**detail**
Validity period starts	Start time that the certificate is valid, in the format ***yyyy mon dd*, *hh*:*mm*:*ss* GMT**.	**detail**
Validity period ends	End time that the certificate is valid, in the format ***yyyy mon dd*, *hh*:*mm*:*ss* GMT**.	**detail**
Alternative name information	Auxiliary identity for the certificate: ***dns-name***, ***email-address***, ***ip-address***, or ***uri*** (uniform resource identifier).	**detail**
Issuer	Information about the entity that has signed and issued the CRL as described in RFC 2459, *Internet X.509 Public Key Infrastructure Certificate and CRL Profile.*	**detail**

Sample Output

show ipsec certificates detail

```
user@host> show ipsec certificates detail
Database: Total entries: 3 Active entries: 4 Locked entries: 1
Subject: C=us, O=x
  ID: 5, References: 0, Serial: 22314868
  Flags: Trusted Non-root Crl-issuer
  Validity period starts: 2003 Mar  1st, 01:20:42 GMT
  Validity period ends: 2003 Mar 31st, 01:50:42 GMT
  Alternative name information:
    IP address: 10.20.210.1
  Issuer: C=FI, O=Company-ABC, CN=Company ABC class 2

Subject: C=us, O=x
  ID: 4, References: 0, Serial: 22315496
  Flags: Trusted Non-root Crl-issuer
  Validity period starts: 2003 Mar  1st, 01:21:45 GMT
  Validity period ends: 2003 Mar 31st, 01:51:45 GMT
  Alternative name information:
    IP address: 10.20.210.20
  Issuer: C=FI, O=Company-ABC, CN=Company ABC class 2

Subject: C=FI, O=SSH Company-ABC, CN=Company ABC class 2
  ID: 1, References: 1, Serial: 1538512
  Flags: Trusted Root Non-crl-issuer
```

```
Validity period starts: 2001 Aug  1st, 07:08:32 GMT
Validity period ends: 2004 Aug  1st, 07:08:32 GMT
Alternative name information:
  Email address: certifier-support@ssh.com
Issuer: C=FI, O=Company-ABC, CN=Company ABC class 2
```

show ipsec redundancy

Syntax show ipsec redundancy (interface <es-*fpc/pic/port*> | security association <*sa-name*>)

Release Information Command introduced before Junos OS Release 7.4.

Description (Encryption interface on M Series and T Series routers only) Display information about IPsec redundancy.

Options

interface <es-*fpc/pic/port*>—Display information about all encryption interfaces, or optionally, about a particular encryption interface.

security association <*sa-name*>—Display information about all remote tunnels, or optionally, about a particular remote tunnel.

Required Privilege Level view

Related Documentation

- request ipsec switch on page 99

List of Sample Output

show ipsec redundancy interface on page 149
show ipsec redundancy security-associations on page 149

Output Fields Table 14 on page 148 lists the output fields for the **show ipsec redundancy** command. Output fields are listed in the approximate order in which they appear.

Table 14: show ipsec redundancy Output Fields

Field Name	Field Description
Failure counter	Number of times a PIC switched between primary and backup interfaces, or the number of times the tunnel switched between the primary and remote peers since the software has been activated.
Primary interface '	Name of the interface configured to be the primary interface.
Backup interface	Name of the interface configured to be the backup interface.
State	State of the primary or backup interface can be **Active**, **Offline**, or **Standby**. Both ES PICs are initialized to **Offline**. For primary and remote peers, **State** can be **Active** or **Standby**. Both peers are in a state of **Standby** by default (there is not yet a connection between the two peers).
Security association	Name of the security association.
Local IP	Local IP address.
Primary remote IP	IP address of the configured primary remote peer.
Backup remote IP	IP address of the configured backup remote peer.

Sample Output

show ipsec redundancy interface

```
user@host> show ipsec redundancy interface
    Failure counter: 0
    Primary interface: es-1/3/0, State: Active
    Backup interface : es-1/1/0, State: Standby
```

show ipsec redundancy security-associations

```
user@host> show ipsec redundancy security-associations sa-dynamic
   Security association: sa-dynamic, Failure counter: 0
   Local IP: 4.4.4.4
   Primary remote IP: 4.4.4.5, State: Standby
   Backup remote IP : 3.3.3.3, State: Standby
```

show ipsec security-associations

Syntax show ipsec security-associations
<brief | detail>
<*sa-name*>

Release Information Command introduced before Junos OS Release 7.4.

Description Display information about the IPsec security associations applied to the local or transit traffic stream.

Options **none**—Display standard information about all IPsec security associations.

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

sa-name—(Optional) Display the specified IPsec security association.

Required Privilege Level view

List of Sample Output show ipsec security-associations sa-name on page 152
show ipsec security-associations sa-name detail on page 152

Output Fields Table 15 on page 150 lists the output fields for the **show ipsec security-associations** command. Output fields are listed in the approximate order in which they appear.

Table 15: show ipsec security-associations Output Fields

Field Name	Field Description	Level of Output
Security association	Name of the security association.	All levels
Interface family	Status of the interface family of the security association. If the interface family field is absent, it is a transport mode security association. The interface family can have one of three options: • **Up**—The security association is referenced in the interface family and the interface family is up. • **Down**—The security association is referenced in the interface family and the interface family is down. • **No reference**—The security association is not referenced in the interface family.	All levels
Local gateway	Gateway address of the local system.	All levels
Remote gateway	Gateway address of the remote system.	All levels
Local identity	Prefix and port number of the local end	All levels
Remote identity	Prefix and port number of the remote end.	All levels
Direction	Direction of the security association: **inbound** or **outbound**.	All levels
SPI	Value of the security parameter index.	All levels

Table 15: show ipsec security-associations Output Fields *(continued)*

Field Name	Field Description	Level of Output
AUX-SPI	Value of the auxiliary security parameter index. • When the value is **AH** or **ESP**, **AUX-SPI** is always **0**. • When the value is**AH+ESP**, **AUX-SPI** is always a positive integer.	All levels
State	Status of the security association: • **Installed**—The security association is installed in the security association database. (For transport mode security associations, the value of **State** must always be **Installed**.) • **Not installed**—The security association is not installed in the security association database.	**detail**
Mode	Mode of the security association: • **transport**—Protects single host-to-host protections. • **tunnel**—Protects connections between security gateways.	All levels
Type	Type of security association:. • **manual**—Security parameters require no negotiation. They are static, and are configured by the user. • **dynamic**—Security parameters are negotiated by the IKE protocol. Dynamic security associations are not supported in transport mode.	All levels
Protocol	Protocol supported: • **transport** mode—Supports Encapsulation Security Protocol (**ESP**) or Authentication Header (**AH**). • **tunnel** mode—Supports **ESP** or **AH+ESP**.	All levels
Authentication	Type of authentication used: **hmac-md5-96**, **hmac-sha1-96**, or **None**.	**detail**
Encryption	Type of encryption used: **des-cbc**, **3des-csc**, or **None**.	**detail**
Soft lifetime **Hard lifetime**	(**dynamic** output only) Each lifetime of a security association has two display options, hard and soft, one of which must be present for a dynamic security association. The **hard lifetime** specifies the lifetime of the SA. The **soft lifetime**, which is derived from the hard lifetime, informs the IPsec key management system that the SA is about to expire. This allows the key management system to negotiate a new SA before the hard lifetime expires. • **Expires in** *seconds* **seconds**—Number of seconds left until the security association expires. • **Expires in** *kilobytes* **kilobytes**—Number of kilobytes left until the security association expires.	**detail**
Anti-replay service	State of the service that prevents packets from being replayed: **Enabled** or **Disabled**.	**detail**

Table 15: show ipsec security-associations Output Fields *(continued)*

Field Name	Field Description	Level of Output
Replay window size	Configured size, in packets, of the antireplay service window: **32** or **64**. The antireplay window size protects the receiver against replay attacks by rejecting old or duplicate packets. If the replay window size is **0**, the antireplay service is **disabled**.	**detail**

Sample Output

show ipsec security-associations sa-name

```
user@host> show ipsec security-associations sa-cosmic brief
Security association: sa-cosmic, Interface family: Up
Local gateway: 21.21.1.1, Remote gateway: 21.21.2.1
Local identity: ipv4_subnet(any:0,[0..7]=0.0.0.0/0)
Remote identity: ipv4_subnet(any:0,[0..7]=0.0.0.0/0)
Direction SPI         AUX-SPI     Mode       Type     Protocol
inbound   2908734119  0           tunnel     dynamic  AH
outbound  3494029335  0           tunnel     dynamic  AH
```

show ipsec security-associations sa-name detail

```
user@host> show ipsec security-associations sa-cosmic detail
Security association: sa-cosmic, Interface family: Up

Local gateway: 21.21.1.1, Remote gateway: 21.21.2.1
Local identity: ipv4_subnet(any:0,[0..7]=0.0.0.0/0)
Remote identity: ipv4_subnet(any:0,[0..7]=0.0.0.0/0)
Direction: inbound, SPI: 2908734119, AUX-SPI: 0, State: Installed
Mode: tunnel, Type: dynamic
Protocol: AH, Authentication: hmac-md5-96, Encryption: None
Soft lifetime: Expired
Hard lifetime: Expires in 120 seconds
Anti-replay service: Disabled

Direction: outbound, SPI: 3494029335, AUX-SPI: 0, State: Installed
Mode: tunnel, Type: dynamic
Protocol: AH, Authentication: hmac-md5-96, Encryption: None
Soft lifetime: Expired
Hard lifetime: Expires in 120 seconds
Anti-replay service: Disabled
```

show system certificate

Syntax
show system certificate
<certificate-id>

Release Information
Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
(Encryption interface on M Series, T Series routers, and QFX Series switches only) Display installed certificates signed by the Juniper Networks certificate authority.

Options
none—Display all installed certificates signed by the Juniper Networks certificate authority.

certificate-id—(Optional) Display the details of a particular certificate.

Required Privilege Level
maintenance

List of Sample Output
show system certificate on page 154
show system certificate (QFX Series) on page 154

Output Fields
Table 16 on page 153 lists the output fields for the **show system certificate** command. Output fields are listed in the approximate order in which they appear.

Table 16: show system certificate Output Fields

Field Name	Field Description
Certificate identifier	Unique identifier associated with a certificate. The certificate identifier is the common name of the subject.
Issuer **Subject**	Information about the certificate issuer and the distinguished name (**DN**) of the issuer, respectively: • **Organization**—Name of the owner's organization. • **Organizational unit**—Name of the owner's department. • **Country**—Two-character country code in which the owner's system is located. • **State**—State in the USA in which the owner is using the certificate. • **Locality**—City in which the owner's system is located. • **Common name**—Name of the owner of the certificate. • **E-mail address**—E-mail address of the owner of the certificate.
Validity	When a certificate is valid.
Signature algorithm	Encryption algorithm applied to the installed certificate.
Public key algorithm	Encryption algorithm applied to the public key.

Sample Output

show system certificate

```
user@host> show system certificate
Certificate identifier: Dallas-v3
  Issuer:
Organization: Juniper Networks, Organizational unit: Juniper CA,
Country:  US, State: CA, Locality: Sunnyvale, Common name: Dallas CA,
E-mail address:ca@juniper.net
  Subject:
  Organization: Juniper Networks, Organizational unit: Juniper CA,
Country: US, State: CA, Locality: Sunnyvale, Common name: Dallas-v3,
E-mail address:ca@juniper.net
Validity:
 Not before: Mar 13 03:23:25 2004 GMT
  Not after: Mar 24 03:23:25 2014 GMT
Signature algorithm: sha1WithRSAEncryption
Public key algorithm: dsaEncryption
```

show system certificate (QFX Series)

```
user@host> show system certificate
Certificate identifier: Dallas-v3
  Issuer:
Organization: Juniper Networks, Organizational unit: Juniper CA,
Country:  US, State: CA, Locality: Sunnyvale, Common name: Dallas CA,
E-mail address:ca@juniper.net
  Subject:
  Organization: Juniper Networks, Organizational unit: Juniper CA,
Country: US, State: CA, Locality: Sunnyvale, Common name: Dallas-v3,
E-mail address:ca@juniper.net
Validity:
 Not before: Mar 13 03:23:25 2004 GMT
  Not after: Mar 24 03:23:25 2014 GMT
Signature algorithm: sha1WithRSAEncryption
Public key algorithm: dsaEncryption
```

PART 4

Index

Index

Symbols

A

B

C

D

E

V