JUNIPER NETWORKS | Engineering Simplicity

Junos® OS

Interfaces User Guide for Security Devices

Published
2025-12-11

JUNOS

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

Junos® OS Interfaces User Guide for Security 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 https://support.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

About This Guide

Use this guide to configure and monitor Network, Services, and Special interfaces for Juniper security devices.

Refer to Interfaces Fundamentals for information on serial interfaces.

CHAPTER 1

Overview

IN THIS CHAPTER

Introduction to Interfaces

IN THIS SECTION

Junos OS supports different types of interfaces on which the devices function. The following topics provide information of types of interfaces used on security devices, the naming conventions and how to monitor the interfaces.

Understanding Interfaces

Interfaces act as a doorway through which traffic enters and exits a device. Juniper Networks devices support a variety of interface types:

- Network interfaces—Networking interfaces primarily provide traffic connectivity.
- Services interfaces—Services interfaces manipulate traffic before it is delivered to its destination.
- Special interfaces—Special interfaces include management interfaces, the loopback interface, and the discard interface.

Each type of interface uses a particular medium to transmit data. The physical wires and Data Link Layer protocols used by a medium determine how traffic is sent. To configure and monitor interfaces, you need to understand their media characteristics, as well as physical and logical properties such as IP addressing, link-layer protocols, and link encapsulation.

NOTE: Most interfaces are configurable, but some internally generated interfaces are not configurable.

Network Interfaces

All Juniper Networks devices use network interfaces to make physical connections to other devices. A connection takes place along media-specific physical wires through an I/O card (IOC) in the SRX Series Services Gateway. Networking interfaces primarily provide traffic connectivity.

You must configure each network interface before it can operate on the device. Configuring an interface can define both the physical properties of the link and the logical properties of a logical interface on the link.

Table 1 on page 3 describes network interfaces that are available on SRX Series Firewalls.

Table 1: Network Interfaces

Interface Name	Description
ae	Aggregated Ethernet interface. See "Understanding Aggregated Ethernet Interfaces" on page 137.
at	ATM-over-ADSL or ATM-over-SHDSL WAN interface.
cl	Physical interface for the 3G wireless modem or LTE Mini-PIM. See "Understanding the 3G Wireless Modem Physical Interface" on page 473 and LTE Mini-PIM Overview. Starting with Junos OS Release 15.1X49-D100, SRX320, SRX340, SRX345, and SRX550HM devices support the LTE interface. The dialer interface is used for initiating wireless WAN connections over LTE networks.
dl	Dialer interface for initiating USB modem or wireless WAN connections. See *USB Modem Interface Overview* and LTE Mini-PIM Overview.
fe	Fast Ethernet interface. See "Understanding Ethernet Interfaces" on page 125.
ge	Gigabit Ethernet interface. See "Understanding Ethernet Interfaces" on page 125.
pt	VDSL2 interface. See Example: Configuring VDSL2 Interfaces (Detail).
reth	For chassis cluster configurations only, redundant Ethernet interface. See "Understanding Ethernet Interfaces" on page 125.

Table 1: Network Interfaces *(Continued)*

Interface Name	Description
se	Serial interface (either RS-232, RS-422/499, RS-530, V.35, or X.21). See Serial Interfaces Overview.
wx	WXC Integrated Services Module (ISM 200) interface for WAN acceleration. See the WXC Integrated Services Module Installation and Configuration.
xe	10-Gigabit Ethernet interface. See "Understanding the 2-Port 10-Gigabit Ethernet XPIM" on page 182.

NOTE: The affected interfaces are these: ATM-over-ADSL or ATM-over-SHDSL (at) interface, dialer interface (dl), E1 (also called DS1) WAN interface, E3 (also called DS3) WAN interface, VDSL2 interface (pt), serial interface (se), T1 (also called DS1) WAN interface, T3 (also called DS3) WAN interface. However, starting from Junos OS Release 15.1X49-D40 and onwards, SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices support VDSL2 (pt), serial (se), T1 (t1) , and E1 (e1) interfaces.

Services Interfaces

Services interfaces provide specific capabilities for manipulating traffic before it is delivered to its destination. On Juniper Networks M Series and T Series routing platforms, individual services such as IP-over-IP encapsulation, link services such as multilink protocols, adaptive services such as stateful firewall filters and NAT, and sampling and logging capabilities are implemented by services Physical Interface Cards (PICs). On SRX Series Firewalls, services processing is handled by the Services Processing Card (SPC).

Although the same Junos OS image supports the services features across all routing platforms, on SRX Series Firewalls, services interfaces are not associated with a physical interface. To configure services on these devices, you configure one or more internal interfaces by specifying slot 0, interface carrier 0, and port 0—for example, gr-0/0/0 for GRE.

Table 2 on page 5 describes services interfaces that you can configure on SRX Series Firewalls.

Table 2: Configurable Services Interfaces

Interface Name	Description
gr-0/0/0	Configurable generic routing encapsulation (GRE) interface. GRE allows the encapsulation of one routing protocol inside another routing protocol. Packets are routed to this internal interface, where they are first encapsulated with a GRE packet and then sent. You can create multiple instances of this interface for forwarding encapsulated data to multiple destination addresses by using the default interface as the parent and creating extensions, for example, gr-0/0/0.1, gr-0/0/0.2, and so on. The GRE interface is an internal interface only and is not associated with a physical interface. It is used only for processing GRE traffic. See the Junos OS Services Interfaces Library for Routing Devices for information about tunnel services.
ip-0/0/0	Configurable IP-over-IP encapsulation (IP-IP tunnel) interface. IP tunneling allows the encapsulation of one IP packet inside another IP packet. With IP routing, you can route IP packets directly to a particular address or route the IP packets to an internal interface where they are encapsulated inside an IP-IP tunnel and forwarded to the encapsulating packet's destination address. You can create multiple instances of this interface for forwarding IP-IP tunnel data to multiple destination addresses by using the default interface as the parent and creating extensions, for example, ip-0/0/0.1, ip-0/0/0.2, and so on. The IP-IP interface is an internal interface only and is not associated with a physical interface. It is used only for processing IP-IP tunnel traffic. See the Junos OS Services Interfaces Library for Routing Devices for information about tunnel services.
lsq-0/0/0	Configurable link services queuing interface. Link services include the multilink services MLPPP, MLFR, and Compressed Real-Time Transport Protocol (CRTP). Packets are routed to this internal interface for link bundling or compression. The link services interface is an internal interface only and is not associated with a physical interface. You must configure the interface for it to perform multilink services. **NOTE**: The ls-0/0/0 interface has been deprecated. All multiclass multilink features supported by ls-0/0/0 are now supported by lsq-0/0/0.
lt-0/0/0	Configurable logical tunnel interface that interconnects logical systems on SRX Series Firewalls. See the Logical Systems and Tenant Systems User Guide for Security Devices.

Table 2: Configurable Services Interfaces *(Continued)*

Interface Name	Description
pp0	Configurable PPPoE encapsulation interface. PPP packets being routed in an Ethernet network use PPPoE encapsulation. Packets are routed to this internal interface for PPPoE encapsulation. The PPPoE encapsulation interface is an internal interface only and is not associated with a physical interface. You must configure the interface for it to forward PPPoE traffic. See "Understanding Point-to-Point Protocol over Ethernet" on page 303.
ppd0	Protocol Independent Multicast (PIM) de-encapsulation interface. In PIM sparse mode, the first-hop routing platform encapsulates packets destined for the rendezvous point device. 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 device, 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 PIM with the `[edit protocol pim]` hierarchy to perform PIM de-encapsulation. Use the `show pim interfaces` command to check the status of ppd0 interface.
ppe0	Protocol Independent Multicast (PIM) encapsulation interface. In PIM sparse mode, the first-hop routing platform encapsulates packets destined for the rendezvous point device. 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 device, 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 PIM with the `[edit protocol pim]` hierarchy to perform PIM encapsulation.
st0	Secure tunnel interface used for IPSec VPNs. See the IPsec VPN User Guide for Security Devices.
umd0	Configurable USB modem physical interface. This interface is detected when a USB modem is connected to the USB port on the device. See *USB Modem Configuration Overview*.

Table 2: Configurable Services Interfaces *(Continued)*

Interface Name	Description
mt-0/0/0	Multicast tunnel interface. This interface is automatically generated, but you can configure properties on it if needed.

Table 3 on page 7 describes non-configurable services interfaces for SRX Series Firewalls.

Table 3: Non-Configurable Services Interfaces

Interface Name	Description
gre	Internally generated Generic Routing Encapsulation (GRE) interface created by Junos OS to handle GRE traffic. It is not a configurable interface.
ipip	Internally generated IP-over-IP interface created by Junos OS to handle IP tunnel traffic. It is not a configurable interface.
lsi	Internally generated link services interface created by Junos OS to handle multilink services like MLPPP, MLFR, and CRTP. It is not a configurable interface.
pc-*pim*/0/0	Internally configured interface used by the system as a control path between the WXC Integrated Services Module and the Routing Engine. It is not a configurable interface. See the WX and WXC Series.
pimd	Internally generated Protocol Independent Multicast (PIM) de-encapsulation interface created by Junos OS to handle PIM de-encapsulation. It is not a configurable interface.
pime	Internally generated Protocol Independent Multicast (PIM) encapsulation interface created by Junos OS to handle PIM encapsulation. It is not a configurable interface.
tap	Internally generated interface created by Junos OS to monitor and record traffic during passive monitoring. Packets discarded by the Packet Forwarding Engine are placed on this interface. It is not a configurable interface.
sp-0/0/0	Adaptive services interface. The logical interface sp-*fpc*/*pic*/*port*. 16383 is an internally generated, non-configurable interface for router control traffic.

Special Interfaces

Special interfaces include management interfaces, which are primarily intended for accessing the device remotely, the loopback interface, which has several uses depending on the particular Junos OS feature being configured, and the discard interface.

Table 4 on page 8 describes special interfaces for SRX Series Firewalls.

Table 4: Special Interfaces

Interface Name	Description
`fxp0, fxp1`	On SRX Series Firewalls, the fxp0 management interface is a dedicated port located on the Routing Engine.
`lo0`	Loopback address. The loopback address has several uses, depending on the particular Junos feature being configured.
`dsc`	Discard interface.

Interface Naming Conventions

Each device interface has a unique name that follows a naming convention. If you are familiar with Juniper Networks M Series and T Series routing platforms, be aware that device interface names are similar to but not identical to the interface names on those routing platforms.

The unique name of each network interface identifies its type and location and indicates whether it is a physical interface or an optional logical unit created on a physical interface.

- The name of each network interface has the following format to identify the physical device that corresponds to a single physical network connector:

  ```
  type-slot/pim-or-ioc/port
  ```

- Network interfaces that are fractionalized into time slots include a channel number in the name, preceded by a colon (:):

```
type-slot/pim-or-ioc/port:channel
```

- Each logical interface has an additional logical unit identifier, preceded by a period (.):

```
type-slot/pim-or-ioc/port:<channel>.unit
```

The parts of an interface name are summarized in Table 5 on page 9.

Table 5: Network Interface Names

Name Part	Meaning	Possible Values
type	Type of network medium that can connect to this interface.	ae, at, ei, e3, fe, fxp0, fxp1, ge, lo0, lsq, lt, ppo, pt, sto, t1, t3, xe, and so on.
slot	Number of the chassis slot in which a PIM or IOC is installed.	SRX5600 and SRX5800 devices: The slot number begins at `0` and increases as follows from left to right, bottom to top: • SRX5600 device—Slots 0 to 5 • SRX5800 device—Slots 0 to 5, 7 to 11 SRX3400 and SRX3600 devices: The Switch Fabric Board (SFB) is always `0`. Slot numbers increase as follows from top to bottom, left to right: • SRX3400 devce—Slots 0 to 4 • SRX3600 device—Slots 0 to 6 • SRX4600 device—Slots 0 to 6

Table 5: Network Interface Names *(Continued)*

Name Part	Meaning	Possible Values
pim-or-ioc	Number of the PIM or IOC on which the physical interface is located.	SRX5600 and SRX5800 devices: For 40-port Gigabit Ethernet IOCs or 4-port 10-Gigabit Ethernet IOCs, this number can be 0, 1, 2, or 3. SRX3400, SRX3600, and SRX 4600 devices: This number is always 0. Only one IOC can be installed in a slot.
port	Number of the port on a PIM or IOC on which the physical interface is located.	On SRX5600 and SRX5800 devices: • For 40-port Gigabit Ethernet IOCs, this number begins at 0 and increases from left to right to a maximum of 9. • For 4-port 10-Gigabit Ethernet IOCs, this number is always 0. On SRX3400, SRX3600, and SRX 4600 devices: • For the SFB built-in copper Gigabit Ethernet ports, this number begins at 0 and increases from top to bottom, left to right, to a maximum of 7. For the SFB built-in fiber Gigabit Ethernet ports, this number begins at 8 and increases from left to right to a maximum of 11. • For 16-port Gigabit Ethernet IOCs, this number begins at 0 to a maximum of 15. • For 2-port 10-Gigabit Ethernet IOCs, this number is 0 or 1. Port numbers appear on the PIM or IOC faceplate.
channel	Number of the channel (time slot) on a fractional or channelized T1 or E1 interface.	• On an E1 interface, a value from 1 through 31. The 1 time slot is reserved. • On a T1 interface, a value from 1 through 24.

Table 5: Network Interface Names *(Continued)*

Name Part	Meaning	Possible Values
unit	Number of the logical interface created on a physical interface.	A value from `0` through `16384`. If no logical interface number is specified, unit `0` is the default, but must be explicitly configured. In addition to user-configured interfaces, there are some logical interfaces that are created dynamically. Hence, for Junos OS, the maximum limit for configuring logical interfaces is 2,62,143 (user configured and dynamically created). Based on performance, for each platform, the maximum number of logical interfaces supported can vary.

NOTE: Platform support depends on the Junos OS release in your installation.

Understanding the Data Link Layer

IN THIS SECTION

The Data Link Layer is Layer 2 in the Open Systems Interconnection (OSI) model. The Data Link Layer is responsible for transmitting data across a physical network link. Each physical medium has link-layer specifications for network and link-layer protocol characteristics such as physical addressing, network topology, error notification, frame sequencing, and flow control.

Physical Addressing

Physical addressing is different from network addressing. Network addresses differentiate between nodes or devices in a network, allowing traffic to be routed or switched through the network. In contrast, physical addressing identifies devices at the link-layer level, differentiating between individual devices on the same physical medium. The primary form of physical addressing is the media access control (MAC) address.

Network Topology

Network topology specifications identify how devices are linked in a network. Some media allow devices to be connected by a bus topology, while others require a ring topology. The bus topology is used by Ethernet technologies, which are supported on Juniper Networks devices.

Error Notification

The Data Link Layer provides error notifications that alert higher layer protocols that an error has occurred on the physical link. Examples of link-level errors include the loss of a signal, the loss of a clocking signal across serial connections, or the loss of the remote endpoint on a T1 or T3 link.

Frame Sequencing

The frame sequencing capabilities of the Data Link Layer allow frames that are transmitted out of sequence to be reordered on the receiving end of a transmission. The integrity of the packet can then be verified by means of the bits in the Layer 2 header, which is transmitted along with the data payload.

Flow Control

Flow control within the Data Link Layer allows receiving devices on a link to detect congestion and notify their upstream and downstream neighbors. The neighbor devices relay the congestion information to their higher layer protocols so that the flow of traffic can be altered or rerouted.

Data Link Sublayers

The Data Link Layer is divided into two sublayers: logical link control (LLC) and media access control (MAC). The LLC sublayer manages communications between devices over a single link of a network. This sublayer supports fields in link-layer frames that enable multiple higher layer protocols to share a single physical link.

The MAC sublayer governs protocol access to the physical network medium. Through the MAC addresses that are typically assigned to all ports on a device, multiple devices on the same physical link

can uniquely identify one another at the Data Link Layer. MAC addresses are used in addition to the network addresses that are typically configured manually on ports within a network.

MAC Addressing

A MAC address is the serial number permanently stored in a device adapter to uniquely identify the device. MAC addresses operate at the Data Link Layer, while IP addresses operate at the Network Layer. The IP address of a device can change as the device is moved around a network to different IP subnets, but the MAC address remains the same, because it is physically tied to the device.

Within an IP network, devices match each MAC address to its corresponding configured IP address by means of the Address Resolution Protocol (ARP). ARP maintains a table with a mapping for each MAC address in the network.

Most Layer 2 networks use one of three primary numbering spaces—MAC-48, EUI-48 (extended unique identifier), and EUI-64—which are all globally unique. MAC-48 and EUI-48 spaces each use 48-bit addresses, and EUI-64 spaces use a 64-bit addresses, but all three use the same numbering format. MAC-48 addresses identify network hardware, and EUI-48 addresses identify other devices and software.

The Ethernet and ATM technologies supported on devices use the MAC-48 address space. IPv6 uses the EUI-64 address space.

MAC-48 addresses are the most commonly used MAC addresses in most networks. These addresses are 12-digit hexadecimal numbers (48 bits in length) that typically appear in one of the following formats:

- *MM:MM:MM:SS:SS:SS*
- *MM-MM-MM-SS-SS-SS*

The first three octets (*MM:MM:MM* or *MM-MM-MM*) are the ID number of the hardware manufacturer. Manufacturer ID numbers are assigned by the Institute of Electrical and Electronics Engineers (IEEE). The last three octets (*SS:SS:SS* or *SS-SS-SS*) make up the serial number for the device, which is assigned by the manufacturer. For example, an Ethernet interface card might have a MAC address of `00:05:85:c1:a6:a0`.

Change History Table

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

Release	Description
15.1X49-D100	Starting with Junos OS Release 15.1X49-D100, SRX320, SRX340, SRX345, and SRX550HM devices support the LTE interface. The dialer interface is used for initiating wireless WAN connections over LTE networks.

Physical Interface Properties

IN THIS SECTION

The physical interfaces on security devices affect the transmission of either link-layer signals or the data across the links. The topics below describes the physical properties that include clocking properties, transmission properties, such as the maximum transmission unit (MTU), and encapsulation methods, such as point-to-point encapsulation. The device also provides support for jumbo frames.

Understanding Interface Physical Properties

The physical properties of a network interface are the characteristics associated with the physical link that affect the transmission of either link-layer signals or the data across the links. Physical properties include clocking properties, transmission properties, such as the maximum transmission unit (MTU), and encapsulation methods, such as point-to-point encapsulation.

The default property values for an interface are usually sufficient to successfully enable a bidirectional link. However, if you configure a set of physical properties on an interface, those same properties must be set on all adjacent interfaces to which a direct connection is made.

Table 6 on page 15 summarizes some key physical properties of device interfaces.

Table 6: Interface Physical Properties

Physical Property	Description
`bert-error-rate`	Bit error rate (BER). The error rate specifies the number of bit errors in a particular bit error rate test (BERT) period required to generate a BERT error condition. See "Understanding Bit Error Rate Testing" on page 17.
`bert-period`	Bit error rate test (BERT) time period over which bit errors are sampled. See "Understanding Bit Error Rate Testing" on page 17.
`chap`	Challenge Handshake Authentication Protocol (CHAP). Specifying `chap` enables CHAP authentication on the interface. See "Understanding CHAP Authentication on a PPPoE Interface" on page 321.
`clocking`	Clock source for the link. Clocking can be provided by the local system (internal) or a remote endpoint on the link (external). By default, all interfaces use the internal clocking mode. If an interface is configured to accept an external clock source, one adjacent interface must be configured to act as a clock source. Under this configuration, the interface operates in a loop timing mode, in which the clocking signal is unique for that individual network segment or loop. See "Understanding Interface Clocking" on page 17.
`description`	A user-defined text description of the interface, often used to describe the interface's purpose.
`disable`	Administratively disables the interface.
`encapsulation`	Type of encapsulation on the interface. Common encapsulation types include PPP, Cisco HDLC, and PPP over Ethernet (PPPoE). See "Understanding Physical Encapsulation on an Interface" on page 269.
`fcs`	Frame check sequence (FCS). FCS is an error-detection scheme that appends parity bits to a digital signal and uses decoding algorithms that detect errors in the received digital signal.

Table 6: Interface Physical Properties *(Continued)*

Physical Property	Description
`mtu`	Maximum transmission unit (MTU) size. MTU is the largest size packet or frame, specified in bytes or octets, that can be sent in a packet-based or frame-based network. The TCP uses MTU to determine the maximum size of each packet in any transmission. You can adjust the MTU values at the physical interfaces by using the following command: `set interface` *`interface-name`* `mtu` *`mtu-value`* Sometimes there is a need to reduce the MTU values on interfaces to match the host tap interface MTU otherwise packets are dropped. You can adjust the MTU values by setting the `mtu` option of the `set interfaces [fxp0 | em0 | fab0 | fab1]` command to a value between 256 and 9192. Example: `user@host#` **`set interfaces em0 mtu 1400`** The supported range for configuring an MTU packet size is 256 through 9192 bytes. However, all interfaces do not support 9192 bytes. For more information on the supported interfaces, see "MTU Default and Maximum Values" on page 20.
`no-keepalives`	Disabling of keepalive messages across a physical link. A keepalive message is sent between network devices to indicate that they are still active. Keepalives help determine whether the interface is operating correctly. Except for ATM-over-ADSL interfaces, all interfaces use keepalives by default.
`pap`	Password Authentication Protocol (PAP). Specifying `pap` enables PAP authentication on the interface. See "Understanding CHAP Authentication on a PPPoE Interface" on page 321.
`payload-scrambler`	Scrambling of traffic transmitted out the interface. Payload scrambling randomizes the data payload of transmitted packets. Scrambling eliminates nonvariable bit patterns (strings of all 1s or all 0s) that generate link-layer errors across some physical links.

Understanding Bit Error Rate Testing

In telecommunication transmission, the bit error rate (BER) is the percentage of bits that have errors compared to the total number of bits received in a transmission, usually expressed as 10 to a negative power. For example, a transmission with a BER of 10^{-6} received 1 errored bit in 1,000,000 bits transmitted. The BER indicates how often a packet or other data unit must be retransmitted because of an error. If the BER is too high, a slower data rate might improve the overall transmission time for a given amount of data if it reduces the BER and thereby lowers the number of resent packets.

A bit error rate test (BERT) is a procedure or device that measures the BER for a given transmission. You can configure a device to act as a BERT device by configuring the interface with a bit error rate and a testing period. When the interface receives a BERT request from a BER tester, it generates a response in a well-known BERT pattern. The initiating device checks the BERT-patterned response to determine the number of bit errors.

Understanding Interface Clocking

IN THIS SECTION

Clocking determines how individual routing nodes or entire networks sample transmitted data. As streams of information are received by a device in a network, a clock source specifies when to sample the data. In asynchronous networks, the clock source is derived locally, and synchronous networks use a central, external clock source. Interface clocking indicates whether the device uses asynchronous or synchronous clocking.

NOTE: Because truly synchronous networks are difficult to design and maintain, most synchronous networks are really plesiochronous networks. In a plesiochronous network, different timing regions are controlled by local clocks that are synchronized (with very narrow constraints). Such networks approach synchronicity and are generally known as synchronous networks.

Most networks are designed to operate as asynchronous networks. Each device generates its own clock signal, or devices use clocks from more than one clock source. The clocks within the network are not

synchronized to a single clock source. By default, devices generate their own clock signals to send and receive traffic.

The system clock allows the device to sample (or detect) and transmit data being received and transmitted through its interfaces. Clocking enables the device to detect and transmit the 0s and 1s that make up digital traffic through the interface. Failure to detect the bits within a data flow results in dropped traffic.

Short-term fluctuations in the clock signal are known as . Long-term variations in the signal are known as .

Asynchronous clocking can either derive the clock signal from the data stream or transmit the clocking signal explicitly.

This topic contains the following sections:

Data Stream Clocking

Common in T1 links, data stream clocking occurs when separate clock signals are not transmitted within the network. Instead, devices must extract the clock signal from the data stream. As bits are transmitted across the network, each bit has a time slot of 648 nanoseconds. Within a time slot, pulses are transmitted with alternating voltage peaks and drops. The receiving device uses the period of alternating voltages to determine the clock rate for the data stream.

Explicit Clocking Signal Transmission

Clock signals that are shared by hosts across a data link must be transmitted by one or both endpoints on the link. In a serial connection, for example, one host operates as a clock primary and the other operates as a clock client. The clock primary internally generates a clock signal that is transmitted across the data link. The clock client receives the clock signal and uses its period to determine when to sample data and how to transmit data across the link.

This type of clock signal controls only the connection on which it is active and is not visible to the rest of the network. An explicit clock signal does not control how other devices or even other interfaces on the same device sample or transmit data.

Understanding Frame Check Sequences

IN THIS SECTION

All packets or frames within a network can be damaged by crosstalk or interference in the network's physical wires. The frame check sequence (FCS) is an extra field in each transmitted frame that can be analyzed to determine if errors have occurred. The FCS uses cyclic redundancy checks (CRCs), checksums, and two-dimensional parity bits to detect errors in the transmitted frames.

This topic contains the following sections:

Cyclic Redundancy Checks and Checksums

On a link that uses CRCs for frame checking, the data source uses a predefined polynomial algorithm to calculate a CRC number from the data it is transmitting. The result is included in the FCS field of the frame and transmitted with the data. On the receiving end, the destination host performs the same calculation on the data it receives.

If the result of the second calculation matches the contents of the FCS field, the packet was sent and received without bit errors. If the values do not match, an FCS error is generated, the frame is discarded and the originating host is notified of the error.

Checksums function similarly to CRCs, but use a different algorithm.

Two-Dimensional Parity

On a link that uses two-dimensional parity bits for frame checking, the sending and receiving hosts examine each frame in the total packet transmission and create a parity byte that is evaluated to detect transmission errors.

For example, a host can create the parity byte for the following frame sequence by summing up each column (each bit position in the frame) and keeping only the least-significant bit:

```
Frame 1       0 1 0 1 0 0 1
Frame 2       1 1 0 1 0 0 1
```

```
Frame 3       1 0 1 1 1 1 0
Frame 4       0 0 0 1 1 1 0
Frame 5       0 1 1 0 1 0 0
Frame 6       1 0 1 1 1 1 1

Parity Byte   1 1 1 1 0 1 1
```

If the sum of the bit values in a bit position is even, the parity bit for the position is 0. If the sum is odd, the parity bit is 1. This method is called even parity. Matching parity bytes on the originating and receiving hosts indicate that the packet was received without error.

MTU Default and Maximum Values

The MTU values are by default without any MTU configurations. If the MTU value is set, then the formula `IFF MTU (IP MTU) = IFD MTU (Media MTU) - L2 Overhead` is applicable. See Table 7 on page 20 for default MTU values.

NOTE: For ATM MLPPP irrespective of UIFD MTU, the IP MTU is always 1500 because the IP MTU calculation is based on the LSQ interface. Even if you configure the LSQ family MTU, the IP MTU value cannot exceed 1504.

Table 7: MTU Values for the PIMs

PIM	Default Media MTU (Bytes)	Maximum MTU (Bytes)	Default IP MTU (Bytes)
1-Port Gigabit Ethernet Small Form-Factor Pluggable (SFP) Mini-PIM	1514	9010	1500
1-Port Small Form-Factor Pluggable (SFP) Mini-PIM	1514	1518	1500
DOCSIS Mini-PIM	1504	1504	1500
Serial Mini-PIM	1504	2000	1500

Table 7: MTU Values for the PIMs *(Continued)*

PIM	Default Media MTU (Bytes)	Maximum MTU (Bytes)	Default IP MTU (Bytes)
T1/E1 Mini-PIM	1504	2000	1500
Dual CT1/E1 GPIM	1504	9000	1500
Quad CT1/E1 GPIM	1504	9000	1500
2-Port 10- Gigabit Ethernet XPIM	1514	9192	1500
16-Port Gigabit Ethernet XPIM	1514	9192	1500
24-Port Gigabit Ethernet XPIM	1514	9192	1500
ADSL2+ Mini-PIM (Encapsulation)			
`atm-snap`	1512	1512	1504
`atm-vcmux`	1512	1512	1512
`atm-nlpid`	1512	1512	1508
`atm-cisco-nlpid`	1512	1512	1510
`ether-over-atm-llc`	1512	1512	1488
`atm-ppp-llc`	1512	1512	1506

Table 7: MTU Values for the PIMs *(Continued)*

PIM	Default Media MTU (Bytes)	Maximum MTU (Bytes)	Default IP MTU (Bytes)
atm-ppp-vcmux	1512	1512	1510
atm-mlppp-llc	1512	1512	1500
ppp-over-ether-over-atm-llc	1512	1512	1480
VDSL- Mini-PIM AT mode (Encapsulation)			
atm-snap	1514	1514	1506
atm-vcmux	1514	1514	1514
atm-nlpid	1514	1514	1510
atm-cisco-nlpid	1514	1514	1512
ether-over-atm-llc	1514	1524	1490
atm-ppp-llc	1514	1514	1508
atm-ppp-vcmux	1514	1514	1512
atm-mlppp-llc	1514	1514	1500
ppp-over-ether-over-atm-llc	1514	1514	1482

Table 7: MTU Values for the PIMs *(Continued)*

PIM	Default Media MTU (Bytes)	Maximum MTU (Bytes)	Default IP MTU (Bytes)
VDSL- Mini-PIM PT mode	1514	1514	1500
G.SHDSL Mini-PIM AT mode (Encapsulation)			
`atm-snap`	4482	4482	4470
`atm-vcmux`	4482	4482	4470
`atm-nlpid`	4482	4482	4470
`atm-cisco-nlpid`	4482	4482	4470
`ether-over-atm-llc`	4482	4482	1500
`atm-ppp-llc`	4482	4482	4476
`atm-ppp-vcmux`	4482	4482	4480
`atm-mlppp-llc`	4482	4482	1500
`ppp-over-ether-over-atm-llc`	4482	4482	1492
G.SHDSL Mini-PIM PT mode	1514	1514	1500

Understanding Jumbo Frames Support for Ethernet Interfaces

The Juniper Networks security devices support jumbo frames up to 9192 bytes.

Jumbo frames are Ethernet frames with more than 1500 bytes of payload (maximum transmission unit [MTU]). Jumbo frames can carry up to 9000 bytes of payload.

You configure jumbo frames at the physical interface by using the following command:

```
set interface interface-name mtu mtu-value
```

Example:

```
user@host# set interfaces ge-0/0/0 mtu 9192
```

The supported range for configuring an MTU packet size is 256 through 9192 bytes. However, all interfaces do not support 9192 bytes. For more information on the supported interfaces, see "MTU Default and Maximum Values" on page 20.

Logical Interface Properties

IN THIS SECTION

The logical interfaces can be configured on the security devices and the description is displayed in the output of the show commands. The logical properties of the security devices include protocol families, IP address or addresses associated with the interface, Virtual LAN (VLAN) tagging, and any firewall filters or routing policies.

Understanding Interface Logical Properties

The logical properties of an interface are the characteristics that do not apply to the physical interface or the wires connected to it. Logical properties include:

- Protocol families running on the interface (including any protocol-specific MTUs)
- IP address or addresses associated with the interface. A *logical interface* can be configured with an IPv6 address, IPv4 address, or both. The IP specification requires a unique address on every interface of each system attached to an IP network, so that traffic can be correctly routed. Individual hosts such as home computers must have a single IP address assigned. Devices must have a unique IP address for every interface.
- Virtual LAN (VLAN) tagging
- Any firewall filters or routing policies that are operating on the interface

SEE ALSO

Understanding Virtual LANs | 38

Understanding Protocol Families

IN THIS SECTION

A protocol family is a group of logical properties within an interface configuration. Protocol families include all the protocols that make up a protocol suite. To use a protocol within a particular suite, you must configure the entire protocol family as a logical property for an interface. The protocol families include common and not-so-common protocol suites.

This topic contains the following sections:

Common Protocol Suites

Junos OS protocol families include the following common protocol suites:

- Inet—Supports IP protocol traffic, including OSPF, BGP, and Internet Control Message Protocol (ICMP).
- Inet6—Supports IPv6 protocol traffic, including RIP for IPv6 (RIPng), IS-IS, and BGP.
- ISO—Supports IS-IS traffic.
- MPLS—Supports MPLS.

NOTE: Junos OS security features are flow-based—meaning the device sets up a flow to examine the traffic. Flow-based processing is not supported for ISO or MPLS protocol families.

Other Protocol Suites

In addition to the common protocol suites, Junos protocol families sometimes use the following protocol suites:

- `ccc`—Circuit cross-connect (CCC).
- `mlfr-uni-nni`—Multilink Frame Relay (MLFR) FRF.16 user-to-network network-to-network (UNI NNI).
- `mlfr-end-to-end`—Multilink Frame Relay end-to-end.
- `mlppp`—Multilink Point-to-Point Protocol.
- `tcc`—Translational cross-connect (TCC).
- `tnp`—Trivial Network Protocol. This Juniper Networks proprietary protocol provides communication between the Routing Engine and the device's packet forwarding components. Junos OS automatically configures this protocol family on the device's internal interfaces only.

Understanding IPv4 and IPv6 Protocol Families

IN THIS SECTION

IPv4 addresses are 32-bit numbers that are typically displayed in dotted decimal notation and contains two primary parts: the network prefix and the host number. The topics below describe the following:

- IPv4 Classful Addressing
- IPv4 Classful Addressing
- IPv4 Dotted Decimal Notation
- IPv4 Subnetting
- IPv4 VLSM
- Understanding IPv6
- IPv6 address types and use of the address types in Junos OS RX Series Firewall
- Configuration of IPv6 Protocol Family

Understanding IPv4 Addressing

IN THIS SECTION

IPv4 addresses are 32-bit numbers that are typically displayed in dotted decimal notation. A 32-bit address contains two primary parts: the network prefix and the host number.

All hosts within a single network share the same network address. Each host also has an address that uniquely identifies it. Depending on the scope of the network and the type of device, the address is either globally or locally unique. Devices that are visible to users outside the network (webservers, for example) must have a globally unique IP address. Devices that are visible only within the network must have locally unique IP addresses.

IP addresses are assigned by a central numbering authority that is called the Internet Assigned Numbers Authority (IANA). IANA ensures that addresses are globally unique where needed and has a large address space reserved for use by devices not visible outside their own networks.

This topic contains the following sections:

IPv4 Classful Addressing

To provide flexibility in the number of addresses distributed to networks of different sizes, 4-octet (32-bit) IP addresses were originally divided into three different categories or classes: class A, class B, and class C. Each address class specifies a different number of bits for its network prefix and host number:

- Class A addresses use only the first byte (octet) to specify the network prefix, leaving 3 bytes to define individual host numbers.
- Class B addresses use the first 2 bytes to specify the network prefix, leaving 2 bytes to define host addresses.
- Class C addresses use the first 3 bytes to specify the network prefix, leaving only the last byte to identify hosts.

In binary format, with an `x` representing each bit in the host number, the three address classes can be represented as follows:

```
00000000 xxxxxxxx xxxxxxxx xxxxxxxx (Class A)
00000000 00000000 xxxxxxxx xxxxxxxx (Class B)
00000000 00000000 00000000 xxxxxxxx (Class C)
```

Because each bit (`x`) in a host number can have a 0 or 1 value, each represents a power of 2. For example, if only 3 bits are available for specifying the host number, only the following host numbers are possible:

```
111 110 101 100 011 010 001 000
```

In each IP address class, the number of host-number bits raised to the power of 2 indicates how many host numbers can be created for a particular network prefix. Class A addresses have 2^{24} (or 16,777,216) possible host numbers, class B addresses have 2^{16} (or 65,536) host numbers, and class C addresses have 2^{8} (or 256) possible host numbers.

IPv4 Dotted Decimal Notation

The 32-bit IPv4 addresses are most often expressed in dotted decimal notation, in which each octet (or byte) is treated as a separate number. Within an octet, the rightmost bit represents 2^{0} (or 1), increasing to the left until the first bit in the octet is 2^{7} (or 128). Following are IP addresses in binary format and their dotted decimal equivalents:

```
11010000 01100010 11000000 10101010 = 208.98.192.170
01110110 00001111 11110000 01010101 = 118.15.240.85
00110011 11001100 00111100 00111011 = 51.204.60.59
```

IPv4 Subnetting

Because of the physical and architectural limitations on the size of networks, you often must break large networks into smaller subnetworks. Within a such a subnetted network, each interface requires its own network number and identifying subnet address.

NOTE: The IP routing world has shifted to Classless Inter-Domain Routing (CIDR). As its name implies, CIDR eliminates the notion of address classes and simply conveys a network prefix along with a mask. The mask indicates which bits in the address identify the network (the prefix). This document discusses subnetting in the traditional context of classful IP addresses.

Figure 1 on page 30 shows a network comprised of three subnets.

Figure 1: Subnets in a Network

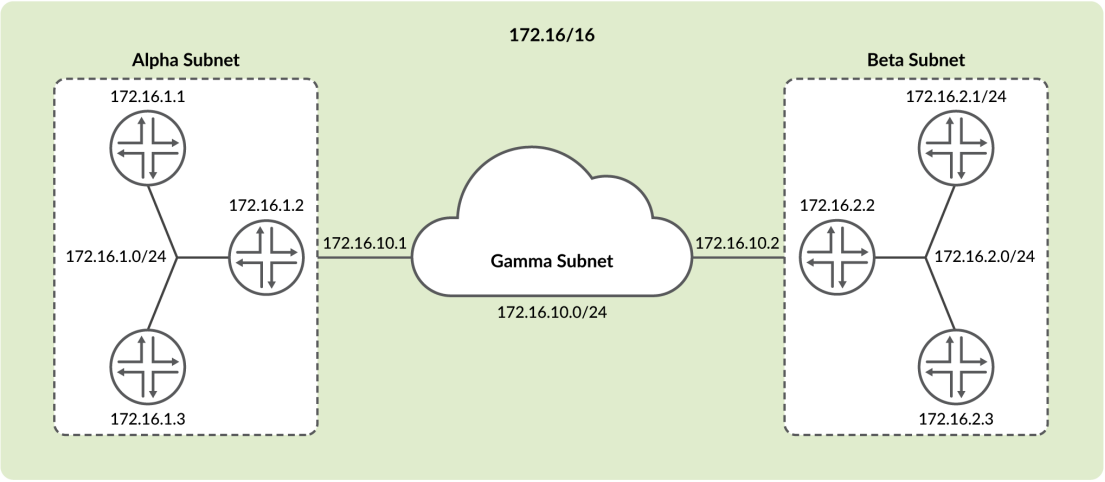


Figure 1 on page 30 shows: three devices connected to the Alpha subnet on the left, three devices connected to the Beta subnet on the right, and a third subnet named Gamma that interconnects the left and right subnets over a WAN link. Collectively, the six devices and three subnets are contained within the larger class B network prefix. In this example, the organization is assigned the network prefix `172.16/16`, which is a class B address. Each subnet is assigned an IP address that falls within this class B network prefix.

In addition to sharing the class B network prefix (the first two octets), each subnet shares the third octet. Because we are using a /24 network mask in conjunction with a class B address, the third octet identifies the subnet. All devices on a subnet must have the same subnet address. In this case, the alpha subnet has the IP address `172.16.1.0/24`, the beta subnet has the IP address `172.16.2.0/24`, and the Gamma subnet is assigned `172.16.10.10/24`.

Taking one of these subnets as an example, the Beta subnet address `172.16.2.0/24` is represented in binary notation as:

```
10101100 . 00010000 . 00000010 . xxxxxxxx
```

Because the first 24 bits in the 32-bit address identify the subnet, the last 8 bits are available to assign to hosts attachments on each subnet. To reference a subnet, the address is written as `172.16.10.0/24` (or just `172.16.10/24`). The `/24` indicates the length of the subnet mask (sometimes written as `255.255.255.0`). This network mask indicates that the first 24 bits identify the network and subnetwork while the last 8 bits identify hosts on the respective subnetwork.

IPv4 Variable-Length Subnet Masks

Traditionally, subnets were divided by address class. Subnets had either 8, 16, or 24 significant bits, corresponding to 2^{24}, 2^{16}, or 2^{8} possible hosts. As a result, an entire /16 subnet had to be allocated for a network that required only 400 addresses, wasting 65,136 (2^{16} – 400 = 65,136) addresses.

To help allocate address spaces more efficiently, variable-length subnet masks (VLSMs) were introduced. Using VLSM, network architects can allocate more precisely the number of addresses required for a particular subnet.

For example, suppose a network with the prefix `192.14.17/24` is divided into two smaller subnets, one consisting of 18 devices and the other of 46 devices.

To accommodate 18 devices, the first subnet must have 2^5 (32) host numbers. Having 5 bits assigned to the host number leaves 27 bits of the 32-bit address for the subnet. The IP address of the first subnet is therefore `192.14.17.128/27`, or the following in binary notation:

```
11000000 . 00001110 . 00010001 . 100xxxxx
```

The subnet mask includes 27 significant digits.

To create the second subnet of 46 devices, the network must accommodate 2^6 (64) host numbers. The IP address of the second subnet is `192.14.17.64/26`, or

```
11000000 . 00001110 . 00010001 . 01xxxxxx
```

By assigning address bits within the larger `/24` subnet mask, you create two smaller subnets that use the allocated address space more efficiently.

Understanding IPv6 Address Space, Addressing, Address Format, and Address Types

IN THIS SECTION

Understanding IP Version 6 (IPv6)

The ongoing expansive growth of the Internet and the need to provide IP addresses to accommodate it—to support increasing numbers of new users, computer networks, Internet-enabled devices, and new and improved applications for collaboration and communication—is escalating the emergent use of a new IP protocol. IPv6, with its robust architecture, was designed to satisfy these current and anticipated near future requirements.

IP version 4 (IPv4) is widely used throughout the world today for the Internet, intranets, and private networks. IPv6 builds upon the functionality and structure of IPv4 in the following ways:

- Provides a simplified and enhanced packet header to allow for more efficient routing.
- Improves support for mobile phones and other mobile computing devices.
- Enforces increased, mandatory data security through IPsec (which was originally designed for it).
- Provides more extensive quality-of-service (QoS) support.

IPv6 addresses consist of 128 bits, instead of 32 bits, and include a scope field that identifies the type of application suitable for the address. IPv6 does not support broadcast addresses, but instead uses multicast addresses for broadcast. In addition, IPv6 defines a new type of address called anycast.

Understanding IPv6 Address Types and How Junos OS for SRX Series Services Gateway Uses Them

IP version 6 (IPv6) includes the following types of addresses:

- Unicast

  A unicast address specifies an identifier for a single interface to which packets are delivered. Under IPv6, the vast majority of Internet traffic is foreseen to be unicast, and it is for this reason that the largest assigned block of the IPv6 address space is dedicated to unicast addressing. Unicast addresses include all addresses other than loopback, multicast, link-local-unicast, and unspecified.

  For SRX Series Firewalls, the flow module supports the following kinds of IPv6 unicast packets:

  - Pass-through unicast traffic, including traffic from and to virtual routers. The device transmits pass-through traffic according to its routing table.
  - Host-inbound traffic from and to devices directly connected to SRX Series interfaces. For example, host-inbound traffic includes logging, routing protocol, and management types of traffic. The flow module sends these unicast packets to the Routing Engine and receives them from it. Traffic is processed by the Routing Engine instead of by the flow module, based on routing protocols defined for the Routing Engine.

The flow module supports all routing and management protocols that run on the Routing Engine. Some examples are OSPFv3, RIPng, TELNET, and SSH.

- Multicast

  A multicast address specifies an identifier for a set of interfaces that typically belong to different nodes. It is identified by a value of 0xFF. IPv6 multicast addresses are distinguished from unicast addresses by the value of the high-order octet of the addresses.

  The devices support only host-inbound and host-outbound multicast traffic. Host inbound traffic includes logging, routing protocols, management traffic, and so on.

- Anycast

  An anycast address specifies an identifier for a set of interfaces that typically belong to different nodes. A packet with an anycast address is delivered to the nearest node, according to routing protocol rules.

  There is no difference between anycast addresses and unicast addresses except for the subnet-router address. For an anycast subnet-router address, the low order bits, typically 64 or more, are zero. Anycast addresses are taken from the unicast address space.

  The flow module treats anycast packets in the same way as it handles unicast packets. If an anycast packet is intended for the device, it is treated as host-inbound traffic, and it delivers it to the protocol stack which continues processing it.

IPv6 Address Scope

Unicast and multicast IPv6 addresses support address scoping, which identifies the application suitable for the address.

Unicast addresses support global address scope and two types of local address scope:

- Link-local unicast addresses—Used only on a single network link. The first 10 bits of the prefix identify the address as a link-local address. Link-local addresses cannot be used outside the link.
- Site-local unicast addresses—Used only within a site or intranet. A site consists of multiple network links. Site-local addresses identify nodes inside the intranet and cannot be used outside the site.

Multicast addresses support 16 different types of address scope, including node, link, site, organization, and global scope. A 4-bit field in the prefix identifies the address scope.

IPv6 Address Structure

Unicast addresses identify a single interface. Each unicast address consists of n bits for the prefix, and 128 – n bits for the interface ID.

Multicast addresses identify a set of interfaces. Each multicast address consists of the first 8 bits of all 1s, a 4-bit flags field, a 4-bit scope field, and a 112-bit group ID:

```
11111111 | flgs | scop | group ID
```

The first octet of 1s identifies the address as a multicast address. The flags field identifies whether the multicast address is a well-known address or a transient multicast address. The scope field identifies the scope of the multicast address. The 112-bit group ID identifies the multicast group.

Similar to multicast addresses, anycast addresses identify a set of interfaces. However, packets are sent to only one of the interfaces, not to all interfaces. Anycast addresses are allocated from the normal unicast address space and cannot be distinguished from a unicast address in format. Therefore, each member of an anycast group must be configured to recognize certain addresses as anycast addresses.

Understanding IPv6 Address Space, Addressing, and Address Types

Addressing is the area where most of the differences between IP version 4 (IPv4) and IPv6 exist, but the changes are largely about the ways in which addresses are implemented and used. IPv6 has a vastly larger address space than the impending exhausted IPv4 address space. IPv6 increases the size of the IP address from the 32 bits that compose an IPv4 address to 128 bits. Each extra bit given to an address doubles the size of the address space.

IPv4 has been extended using techniques such as Network Address Translation (NAT), which allows for ranges of private addresses to be represented by a single public address, and temporary address assignment. Although useful, these techniques fall short of the requirements of novel applications and environments such as emerging wireless technologies, always-on environments, and Internet-based consumer appliances.

In addition to the increased address space, IPv6 addresses differ from IPv4 addresses in the following ways:

- Includes a scope field that identifies the type of application that the address pertains to
- Does not support broadcast addresses, but instead uses multicast addresses to broadcast a packet
- Defines a new type of address, called anycast

Understanding IPv6 Address Format

All IPv6 addresses are 128 bits long, written as 8 sections of 16 bits each. They are expressed in hexadecimal representation, so the sections range from 0 to FFFF. Sections are delimited by colons, and leading zeroes in each section may be omitted. If two or more consecutive sections have all zeroes, they can be collapsed to a double colon.

IPv6 addresses consist of 8 groups of 16-bit hexadecimal values separated by colons (:). IPv6 addresses have the following format:

```
aaaa:aaaa:aaaa:aaaa:aaaa:aaaa:aaaa:aaaa
```

Each `aaaa` is a 16-bit hexadecimal value, and each `a` is a 4-bit hexadecimal value. Following is a sample IPv6 address:

```
3FFE:0000:0000:0001:0200:F8FF:FE75:50DF
```

You can omit the leading zeros of each 16-bit group, as follows:

```
3FFE:0:0:1:200:F8FF:FE75:50DF
```

You can compress 16-bit groups of zeros to double colons (::) as shown in the following example, but only once per address:

```
3FFE::1:200:F8FF:FE75:50DF
```

An IPv6 address prefix is a combination of an IPv6 prefix (address) and a prefix length. The prefix takes the form *ipv6-prefix*/*prefix-length* and represents a block of address space (or a network). The *ipv6-prefix* variable follows general IPv6 addressing rules. The *prefix-length* variable is a decimal value that indicates the number of contiguous, higher-order bits of the address that make up the network portion of the address. For example, 10FA:6604:8136:6502::/64 is a possible IPv6 prefix with zeros compressed. The site prefix of the IPv6 address 10FA:6604:8136:6502::/64 is contained in the left most 64 bits, 10FA:6604:8136:6502.

For more information on the text representation of IPv6 addresses and address prefixes, see RFC 4291, *IP Version 6 Addressing Architecture*.

Use Feature Explorer to confirm platform and release support for specific features.

Review the section "Understanding IPv6 Address Space, Addressing, Address Format, and Address Types" on page 31for notes related to your platform

Platform-Specific IPv6 Address Format Behavior

Use Feature Explorer to confirm platform and release support for specific features.

Use the following table to review platform-specific behavior for your platform:

Platform	Difference
SRX Series	• SRX300, SRX320, SRX340, SRX345, and SRX380 devices that support IPv6 Address Format, the changes in source AS and destination AS are not immediately reflected in exported flows. • SRX300, SRX320, SRX340, SRX345, and SRX380 devices that support IPv6 Address Format, IPv6 traffic transiting over IPv4 based IP over IP tunnel (for example, IPv6-over-IPv4 using ip-x/x/x interface) is not supported.

SEE ALSO

About the IPv6 Basic Packet Header

Configuring the inet6 IPv6 Protocol Family

In configuration commands, the protocol family for IPv6 is named `inet6`. In the configuration hierarchy, instances of `inet6` are parallel to instances of `inet`, the protocol family for IPv4. In general, you configure `inet6` settings and specify IPv6 addresses in parallel to `inet` settings and IPv4 addresses.

NOTE: On SRX Series Firewalls, on configuring identical IPs on a single interface, you will not see a warning message; instead, you will see a syslog message.

The following example shows the CLI commands you use to configure an IPv6 address for an interface:

```
[edit]
user@host# show interfaces
ge-0/0/0 {
    unit 0 {
        family inet {
            address 10.100.37.178/24;
        }
    }
```

```
}

[edit]
user@host# set interfaces ge-0/0/0 unit 0 family ?
Possible completions:
+ apply-groups         Groups from which to inherit configuration data
+ apply-groups-except  Don't inherit configuration data from these groups
> ccc                  Circuit cross-connect parameters
> ethernet-switching   Ethernet switching parameters
> inet                 IPv4 parameters
> inet6                IPv6 protocol parameters
> iso                  OSI ISO protocol parameters
> mpls                 MPLS protocol parameters
> tcc                  Translational cross-connect parameters
> vpls                 Virtual private LAN service parameters

[edit]
user@host# set interfaces ge-0/0/0 unit 0 family inet6 address 8d8d:8d01::1/64
user@host# show interfaces
ge-0/0/0 {
    unit 0 {
        family inet {
            address 10.100.37.178/24;
        }
        family inet6 {
            address 8d8d:8d01::1/64;
        }
    }
}
```

SEE ALSO

- Enabling Flow-Based Processing for IPv6 Traffic

Configuring VLAN Tagging

IN THIS SECTION

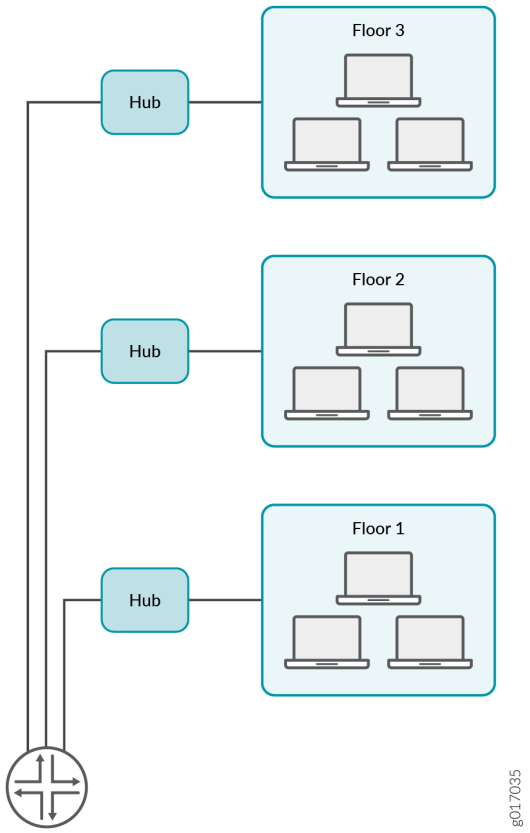
Virtual LANs (VLANs) allow network architects to segment LANs into different broadcast domains based on logical groupings. The topic below describes the configuration of these tagged VLANs, VLAN IDs, and supported Ethernet interface types.

Understanding Virtual LANs

A LAN is a single broadcast domain. When traffic is broadcast, all hosts within the LAN receive the broadcast traffic. A LAN is determined by the physical connectivity of devices within the domain.

Within a traditional LAN, hosts are connected by a hub or repeater that propagates any incoming traffic throughout the network. Each host and its connecting hubs or repeaters make up a LAN segment. LAN segments are connected through switches and bridges to form the broadcast domain of the LAN. Figure 2 on page 39 shows a typical LAN topology.

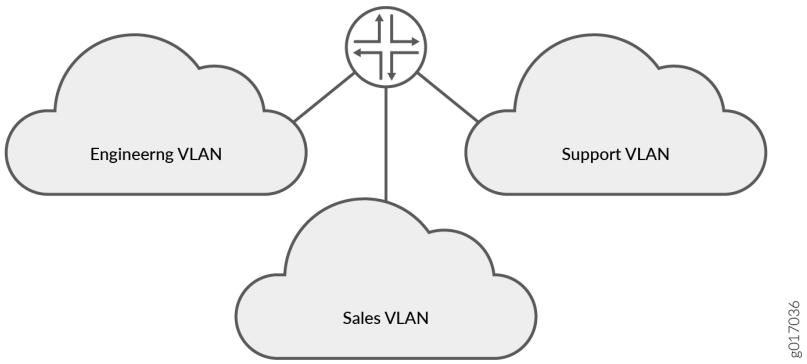
Figure 2: Typical LAN



Virtual LANs (VLANs) allow network architects to segment LANs into different broadcast domains based on logical groupings. Because the groupings are logical, the broadcast domains are not determined by the physical connectivity of the devices in the network. Hosts can be grouped according to a logical function, to limit the traffic broadcast within the VLAN to only the devices for which the traffic is intended.

Suppose a corporate network has three major organizations: engineering, sales, and support. Using VLAN tagging, hosts within each organization can be tagged with a different VLAN identifier. Traffic sent to the broadcast domain is then checked against the VLAN identifier and broadcast to only the devices in the appropriate VLAN. Figure 3 on page 40 shows a typical VLAN topology.

Figure 3: Typical VLAN



SEE ALSO

MPLS Applications User Guide

VLAN IDs and Ethernet Interface Types Supported

The below table lists VLAN ID range by interface type supported:

Table 8: VLAN ID Range by Interface Type Supported

Interface Type	Interface Type VLAN ID Range
2-Port 10-Gigabit Ethernet	1 through 4094
10-Gigabit Ethernet	1 through 4094
16-Port Gigabit Ethernet	1 through 4094
24-Port Gigabit Ethernet	1 through 4094
Aggregated Ethernet for Fast Ethernet	1 through 1023

Table 8: VLAN ID Range by Interface Type Supported *(Continued)*

Interface Type	Interface Type VLAN ID Range
Aggregate Ethernet for Gigabit Ethernet	1 through 4094
Gigabit Ethernet	1 through 4094
Management and internal Ethernet interfaces	1 through 1023

SEE ALSO

Understanding Interface Physical Properties | 14

Configuring VLAN Tagging

IN THIS SECTION

Use Feature Explorer to confirm platform and release support for specific features.

Review the "Platform-Specific VLAN Behavior" on page 45 section for notes related to your platform.

See Table 9 on page 42 for flexible VLANs.

Table 9: Flexible VLANs

Number of Tags	VLAN ID
0 (Untagged)	Native
1 (Tagged)	Single
2 (Dual tagged)	Dual

This topic includes the following sections:

Configuring Single-Tag Framing

To configure a device to receive and forward single-tag frames with 802.1Q VLAN tags, include the `vlan-tagging` statement at the `[edit interfaces` *`interface-name`*`]` hierarchy level:

```
[edit interfaces interface-name]
vlan-tagging;
```

Configuring Dual Tagging

To configure the device to receive and forward dual-tag frames with 802.1Q VLAN tags, include the `flexible-vlan-tagging` statement at the `[edit interfaces` *`interface-name`*`]` hierarchy level:

```
[edit interfaces interface-name]
flexible-vlan-tagging;
```

Configuring Mixed Tagging

To configure mixed tagging, include the `flexible-vlan-tagging` statement at the `[edit interfaces ge-`*`fpc/pic/port`*` ]` hierarchy level. You must also include the `vlan-tags` statement with `inner` and `outer` options or the `vlan-id` statement at the `[edit interfaces ge-`*`fpc/pic/port`*` unit` *`logical-unit-number`*`]` hierarchy level:

```
[edit interfaces ge-fpc/pic/port]
flexible-vlan-tagging;
unit logical-unit-number {
```

```
    vlan-id number;
    family family {
        address address;
    }
}
unit logical-unit-number {
    vlan-tags inner tpid.vlan-id outer tpid.vlan-id;
    family family {
        address address;
    }
}
```

NOTE: When you configure the physical interface MTU for mixed tagging, you must increase the MTU to 4 bytes more than the MTU value you would configure for a standard VLAN-tagged interface.

For example, if the MTU value is configured to be 1018 on a VLAN-tagged interface, then the MTU value on a flexible VLAN tagged interface must be 1022—4 bytes more. The additional 4 bytes accommodates the future addition of a stacked VLAN tag configuration on the same physical interface.

The following example configures mixed tagging. Dual-tag and single-tag logical interfaces are under the same physical interface:

```
[edit interfaces ge-0/2/0]
flexible-vlan-tagging;
unit 0 {
    vlan-id 232;
    family inet {
        address 10.66.1.2/30;
    }
}
unit 1 {
    vlan-tags outer 0x8100.222 inner 0x8100.221;
    family inet {
        address 10.66.1.2/30;
    }
}
```

Configuring Mixed Tagging Support for Untagged Packets

You can configure mixed tagging support for untagged packets on a port. Untagged packets are accepted on the same mixed VLAN-tagged port. To accept untagged packets, include the `native-vlan-id` statement and the `flexible-vlan-tagging` statement at the `[edit interfaces` *`interface-name`*`]` hierarchy level:

```
[edit interfaces ge-fpc/pic/port]
flexible-vlan-tagging;
native-vlan-id number;
```

NOTE: The `flexible-vlan-tagging` is supported only with either no encapsulation or VPLS VLAN encapsulation.

The *logical interface* on which untagged packets are to be received must be configured with the same native VLAN ID as that configured on the physical interface. To configure the logical interface, include the `vlan-id` statement (matching the `native-vlan-id` statement on the physical interface) at the `[edit interfaces` *`interface-name`* `unit` *`logical-unit-number`*`]` hierarchy level.

The following example configures untagged packets to be mapped to logical unit number 0:

```
[edit interfaces ge-0/2/0]
flexible-vlan-tagging;
native-vlan-id 232;
unit 0 {
    vlan-id 232;
    family inet {
        address 10.66.1.2/30;
    }
}
unit 1 {
    vlan-tags outer 0x8100.222 inner 0x8100.221;
    family inet {
        address 10.66.1.2/30;
    }
}
```

Platform-Specific VLAN Behavior

Use Feature Explorer to confirm platform and release support for specific features.

Use the following table to review platform-specific VLAN configuration behavior for your platform:

Platform	Difference
SRX Series	• On SRX320 and SRX340 devices, when using the 1-GE SFP Mini-PIM, the VLAN ID 4093 falls within the reserved VLAN address range. Because of this, you cannot configure VLAN ID 4093 on these platforms. Platform support for this restriction may vary depending on the Junos OS release in use. • On SRX300, SRX320, SRX340, SRX345, and SRX380 devices, you can configure the system to receive and forward single-tag frames, dual-tag frames, or a mixture of single-tag and dual-tag frames. • On SRX5400, SRX5600, and SRX5800 devices support single-tag framing. • On SRX300, SRX320, SRX340, SRX345, and SRX380 devices, mixed tagging is supported, allowing you to configure two logical interfaces on the same Ethernet port—one with single-tag framing and the other with dual-tag framing.

2

CHAPTER

Configuring ADSL and SHDSL Interfaces

IN THIS CHAPTER

ADSL and SHDSL Interfaces

SUMMARY

Learn about ADSL and SHDSL interface details and how to configure the interfaces on security devices.

IN THIS SECTION

ADSL and SHDSL Interface Overview

Asymmetric digital subscriber line (ADSL) technology is part of the *x*DSL family of modem technologies that uses existing twisted-pair telephone lines to transport high-bandwidth data. The Symmetric high-speed DSL (SHDSL) interfaces support an SHDSL multirate technology which helps in data transfer between a single CPE subscriber and a central office (CO). The G.SHDSL Mini-*Physical Interface Module* (Mini-PIM) provides the physical connection to DSL network media types. Table 10 on page 47 specifies the key details of the ADSL, SHDSL interfaces, and G.SHDSL Mini-PIM.

Table 10: ADSL and SHDSL Interface Details

Interface Details	Description
Interface name	ADSL, SHDSL
Supported on	For information about platforms support, see hardware compatibility tool (HCT).
Interface type	• `at-` represents ADSL2, SHDSL interface and G.SHDSL Mini-PIM when you configure `at-` to function as VDSL2.

Table 10: ADSL and SHDSL Interface Details *(Continued)*

Interface Details	Description
ADSL/ADSL2/ ADSL2+ use cases	• Connects the loop between service provider networks and customer sites. ADSL Mini-PIM facilitates a maximum of 10 virtual circuits on supported security devices and can use PPP over Ethernet over ATM (PPPoEoA) and PPP over ATM (PPPoA) to connect through ADSL lines only. • Modems work as a dual-purpose ADSL circuit and can accommodate lower-frequency voice traffic and higher-frequency data traffic. • Improves the data rate and reach performance, diagnostics, standby mode, and interoperability of ADSL modems. It doubles the possible downstream data bandwidth, enabling rates of 20 Mbps on telephone lines shorter than 5000 feet (1.5 km). • Uses Seamless Rate Adaptation (SRA) to change the data rate of a connection during operation with no interruptions or bit errors and the ADSL2 transceiver detects changes in channel conditions with data transmission parameters.
SHDSL use cases	• Supports an SHDSL multirate technology for data transfer between a single customer premises equipment (CPE) subscriber and a central office (CO). ITU-T G.991.2 is the official standard for describing SHDSL, also known as G.SHDSL. • Delivers a bandwidth of up to 2.3 Mbps in symmetrical directions. Compatible with ADSL and therefore causes very little, if any, interference between cables and is deployed on a network similar to ADSL.
GSHDSL Mini-PIM use cases	Provides the physical connection to DSL network media types and extended ATM CoS functionality to cells across the network. By default, unspecified bit rate (UBR) is used because the bandwidth utilization is unlimited. You can define bandwidth utilization with sustained cell rate and burst tolerance.

For information on ADSL2 hardware specifications, see 1-Port ADSL2+ Mini-Physical Interface Module Network Interface Specifications.

Features Supported on the ADSL, ADSL2, and SHDSL Interface

Table 11 on page 49 describes the key features supported on ADSL2 and SHDSL interfaces.

Table 11: Key Features Supported on ADSL2 and SHDSL

Feature	Description
ADSL Features	
DSL	• Supports ATM-over-ADSL and ATM-over-SHDSL interfaces. Payload loopback functionality is not supported on ATM-over-SHDSL interfaces. • Uses PPP over Ethernet over ATM (PPPoEoA) and PPP over ATM (PPPoA) to connect through ADSL lines only for supported security devices with Mini-PIMs.
ATM CoS Support	Ability of a network to guarantee *class of service* depends on the way in which the source generates cells and on the availability of network resources. Based on the way in which the source generates cells and the availability of network resources, the set of traffic descriptors specified are: • Peak cell rate (PCR)—Top rate at which traffic can burst. • Sustained cell rate (SCR)—Normal traffic rate averaged over time. • Maximum burst size (MBS)—Maximum burst size that can be sent at the peak rate. • Cell delay variation tolerance (CDVT)—Allows you to delay the traffic for a particular time duration in microseconds to follow a rhythmic pattern.
Encapsulation	You can enable an existing Junos OS CLI to support MLPPP encapsulation and the family mlppp. To establish an ADSL link, you must first use an RJ-11 cable to connect the CPE to a DSLAM patch panel to form an ADSL link and then use OC3 or DS3 to connect the DSLAM to M Series or E Series devices to form an ATM backbone.
SHDSL Features	
Bandwidth	SHDSL is symmetrical and delivers a bandwidth of up to 2.3 Mbps in both directions. Compatible with ADSL and therefore causes very little, if any, interference between cables.

Table 11: Key Features Supported on ADSL2 and SHDSL *(Continued)*

Feature	Description
Packet Transfer Mode (PTM)	Supports PTM and packets (IP, PPP, Ethernet, MPLS, and so on) are transported over DSL links as an alternative to using Asynchronous Transfer Mode (ATM). PTM is based on the Ethernet in the First Mile (EFM) IEEE 802.3ah standard.
DSL	G.SHDSL Mini-*Physical Interface Module* (Mini-PIM) provides the physical connection to DSL network media types.
GSHDSL Virtual circuits (VC)	VC per Mini-PIM (10 maximum including OAM VC).
MTU size	Maximum MTU size of 9180 bytes.
GSHDSL PTM EFM	• Supports EFM PIC mode, PPPoE encapsulation, IPv6, *Chassis cluster* mode, and VLAN over EFM. • Maximum MTU size of 1514 bytes.

For more information on supported features and profiles on ADSL2 interfaces, see 1-Port ADSL2+ Mini-Physical Interface Module Key Features and for SHDSL and GSHDSL interfaces, see 1-Port G.SHDSL 8-Wire Mini-Physical Interface Module Overview.

ADSL transmission is asymmetric because the downstream bandwidth is typically greater than the upstream bandwidth. The typical bandwidths of ADSL, ADSL2, and ADSL2+ circuits are defined in Table 12 on page 50.

Table 12: Standard Bandwidths of DSL Operating Modes

Operating Modes	Upstream	Downstream
ADSL	800 Kbps—1Mbps	8 Mbps
ADSL2	1—1.5 Mbps	12—14 Mbps
ADSL2+	1—1.5 Mbps	24—25 Mbps

Table 12: Standard Bandwidths of DSL Operating Modes *(Continued)*

Operating Modes	Upstream	Downstream
ADSL2+ Annex M	2.5—3 Mbps	25 Mbps

Operating Modes and Line Rates of the G.SHDSL Mini-PIM

The G.SHDSL Mini-PIM supports 2-wire (4-port 2-wire) mode, 4-wire (2-port 4-wire) mode, 8-wire (1-port 8-wire) mode, and EFM mode. The default operating mode is 2x 4-wire for this G.SHDSL Mini-PIM. G.SHDSL is supported on all devices using the symmetrical WAN speeds shown in Table 13 on page 51.

Table 13: Symmetrical WAN Speeds

Modes	Symmetrical WAN Speed Using Annex A and B	Symmetrical WAN Speed Using Annex F and G
2-wire	2.3 Mbps	From 768 Kbps to 5.696 Mbps
4-wire	4.6 Mbps	From 1.536 Mbps to 11.392 Mbps
8-wire	9.2 Mbps	From 3.072 Mbps to 22.784 Mbps
EFM mode	2.3 Mbps	From 768 Kbps to 5.696 Mbps
NOTE: A maximum of 16 Mbps is supported on SRX210, SRX220, SRX240, and SRX550 devices.		

Example: Configure ADSL and SHDSL Network Interfaces

In this example you configure the ADSL and SHDSL interface on an SRX Series Firewall which supports LFI through an MLPPP. To support MLPPP encapsulation and the family mlppp on the ADSL interface on an SRX Series Firewall, you enable an existing Junos OS CLI. To establish an ADSL link between network devices, you must use some intermediate connections. First, use an RJ-11 cable to connect the CPE (for example, an SRX Series Firewall) to a DSLAM patch panel to form an ADSL link. Then use OC3 or DS3 to connect the DSLAM to M Series or E Series devices to form an ATM backbone.

Table 14 on page 52 specifies the CLI quick configuration commands used for configuring ADSL and SHDSL interfaces.

Table 14: CLI Quick Configuration

Configuration Step	CLI Quick Configuration Commands
Configure the DHCP client on ADSL interface	set interfaces at-1/0/0 encapsulation ethernet-over-atm set interfaces at-1/0/0 atm-options vpi 2 set interfaces at-1/0/0 dsl-options operating-mode auto set interfaces at-1/0/0 unit 0 set interfaces at-1/0/0 unit 0 encapsulation ether-over-atm-llc set interfaces at-1/0/0 unit 0 vci 2.122 set interfaces at-1/0/0 unit 0 family inet set interfaces at-1/0/0 unit 0 family inet dhcp
Configure the IPv6 address on an ADSL interface	set interfaces at-1/0/0 encapsulation ethernet-over-atm set interfaces at-1/0/0 atm-options vpi 2 set interfaces at-1/0/0 unit 0 encapsulation ether-over-atm-llc set interfaces at-1/0/0 unit 0 vci 2.118 set interfaces at-1/0/0 unit 0 family inet6 address 13:13::1/64
Configure ATM-over-ADSL network interfaces	set interfaces at-2/0/0 atm-options vpi 25 oam-liveness up-count 200 down-count 200 set interfaces at-2/0/0 atm-options vpi 25 oam-period 100 set interfaces at-1/0/0 unit 0 shaping cbr set interfaces at-1/0/0 unit 0 shaping vbr peak 33000 set interfaces at-1/0/0 dsl-options operating-mode auto set interfaces at-1/0/0 encapsulation ethernet-over-atm set interfaces at-1/0/0 unit 3 encapsulation atm-nlpid oam-liveness up-count 200 down-count 200 set interfaces at-1/0/0 unit 3 oam-period 100 set interfaces at-1/0/0 unit 3 family inet set interfaces at-1/0/0 unit 3 vci 35
Configure CHAP on DSL interfaces	set access profile A-ppp-client client client1 chap-secret my-secret set interfaces at-3/0/0 unit 0 ppp-options chap access-profile A-ppp-client local-name A-at-3/0/0.0 passive

Table 14: CLI Quick Configuration *(Continued)*

Configuration Step	CLI Quick Configuration Commands
Configure ATM-over-SHDSL network interfaces	**set chassis fpc 6 pic 0 shdsl pic-mode 1-port-atm** **set interfaces at-2/0/0 atm-options vpi 25 oam-liveness up-count 200 down-count 200** **set interfaces at-2/0/0 atm-options vpi 25 oam-period 100** **set interfaces at-2/0/0 encapsulation ethernet-over-atm shdsl-options annex annex-a** **set interfaces at-2/0/0 encapsulation ethernet-over-atm shdsl-options line-rate auto** **set interfaces at-2/0/0 encapsulation ethernet-over-atm shdsl-options loopback local** **set interfaces at-2/0/0 encapsulation ethernet-over-atm shdsl-options snr-margin current 5 snext 5** **set interfaces at-2/0/0 unit 3 encapsulation atm-nlpid** **set interfaces at-2/0/0 unit 3 oam-liveness up-count 200 down-count 200** **set interfaces at-2/0/0 unit 3 oam-period 100** **set interfaces at-2/0/0 unit 3 oam-period 100** **set interfaces at-2/0/0 unit 3 vci 35**

Configure the DHCP client on ADSL interface

In this example, you configure the ATM interface as `at-1/0/0`. Then set the logical interface to `unit 0` and specify the family protocol type as `inet`. Finally, configure the DHCP client. To configure DHCP client on ADSL interfaces:

1. Set the encapsulation mode.

```
[edit]
user@host# set interfaces at-1/0/0 encapsulation ethernet-over-atm
```

2. Configure the ATM VPI option.

```
[edit]
user@host# set interfaces at-1/0/0 atm-options vpi 2
```

3. Set operating mode.

```
[edit]
user@host# set interfaces at-1/0/0 dsl-options operating-mode auto
```

4. Set the logical interface.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0
```

5. Set the encapsulation mode for logical interface.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 encapsulation ether-over-atm-llc
```

6. Set the ATM VCI option.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 vci 2.122
```

7. Specify the family protocol type.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 family inet
```

8. Configure the DHCP client.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 family inet dhcp
```

9. Set the DHCP client identifier as a ASCII or hexadecimal value (optional):

   Use hexadecimal if the client identifier is a MAC address—for example, 00:0a:12:00:12:12.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 family inet dhcp client-identifier
00:0a:12:00:12:12
```

10. Set the DHCP lease time in seconds—for example, 86400 (24 hours). The range is 60 through 2147483647 seconds (optional).

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 family inet dhcp lease-time 86400
```

11. Define the number of attempts allowed to retransmit a DHCP packet (optional)—for example, 6

    The range is 0 through 6. The default is 4 times.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 family inet dhcp retransmission-attempt 6
```

12. Define the interval, in seconds, allowed between retransmission attempts (optional)—for example, 5.

    The range is 4 through 64. The default is 4 seconds.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 family inet dhcp retransmission-interval 5
```

13. Set the IPv4 address of the preferred DHCP server (optional)—for example, 10.1.1.1.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 family inet dhcp server-address 10.1.1.1
```

14. Set the vendor class ID for the DHCP client (optional)—for example, ether.

```
[edit]
user@host# set interfaces at-0/0/1 unit 0 family inet dhcp vendor-id ether
```

Use the `show interfaces at-1/0/0` command to see the output of the configuration.

Configure the IPv6 Address on an ADSL Interface

To configure the IPv6 address on an ADSL interface:

1. Configure the encapsulation type.

```
[edit]
user@host# set interfaces at-1/0/0 encapsulation ethernet-over-atm
```

2. Specify the annex type.

```
[edit]
user@host# set interfaces at-1/0/0 atm-options vpi 2
```

3. Configure the encapsulation for the logical unit.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 encapsulation ether-over-atm-llc
```

4. Configure the VCI value.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 vci 2.118
```

5. Configure family protocol type and assign an IPv6 address.

```
[edit]
user@host# set interfaces at-1/0/0 unit 0 family inet6 address 13:13::1/64
```

Use the `show interfaces at-1/0/0` command to see the output of the configuration.

Configure ATM-over-ADSL Network Interfaces

This example shows how to use devices with ADSL Annex A or Annex B PIMs to send network traffic through a point-to-point connection to a DSLAM. Within the example, you set the DSL operating mode type to auto so that the ADSL interface will autonegotiate settings with the DSLAM.

The example shows how to create an ATM interface called at-2/0/0. The values for the interface's physical properties are kept relatively low—the ATM VPI is set to 25; both the OAM down count and up count are set to 200 cells; the OAM period is set to 100 seconds.

The example also shows how to set traffic shaping values on the ATM interface to support CoS. CBR is enabled in order to stabilize the cell transmission rate throughout the duration of the connection. Additionally, the VBR peak is set to 33,000 for data packet transfers.

Within the example, you set the encapsulation mode to ethernet-over-atm to support PPP over Ethernet IPv4 traffic. You also configure a logical interface (unit 3). The logical interface uses ATM NLPID encapsulation. As with the physical interface, the OAM down count and up count are set to 200 cells on the logical interface and the OAM period is set to 100 seconds. The family protocol is set to inet and the VCI is set to 35.

On SRX Series Firewalls, the ATM interface takes more than 5 minutes to come up when CPE is configured in ANSI-DMT mode and CO is configured in automode. This occurs only with ALU 7300 DSLAM, due to limitation in current firmware version running on the ADSL Mini-PIM.

To configure ATM-over-ADSL network interfaces for the devices:

1. Create an ATM interface.

```
[edit]
user@host# edit interfaces at-2/0/0
```

2. Configure the physical properties for the ATM interface.

```
[edit interfaces at-2/0/0]
user@host# set atm-options vpi 25
user@host# set atm-options vpi 25 oam-liveness up-count 200 down-count 200
user@host# set atm-options vpi 25 oam-period 100
```

3. Specify the CBR value and VBR value for the Ethernet interface.

```
[edit]
user@host# edit interfaces at-1/0/0 unit 0
user@host# set shaping cbr
user@host# set shaping vbr peak 33000
```

4. Set the DSL operating mode type.

```
[edit interfaces at-1/0/0.0]
user@host# set dsl-options operating-mode auto
```

5. Configure the encapsulation type.

```
[edit interfaces at-1/0/0]
user@host# set encapsulation ethernet-over-atm
```

6. Configure the encapsulation for the logical unit.

```
[edit interfaces at-1/0/0 unit 3]
user@host# set encapsulation atm-nlpid
```

7. Configure the OAM liveness values for an ATM virtual circuit.

```
[edit interfaces at-1/0/0 unit 3]
user@host# set oam-liveness up-count 200 down-count 200
```

8. Specify the OAM period.

```
[edit interfaces at-1/0/0 unit 3]
user@host# set oam-period 100
```

9. Set the family protocol type.

```
[edit interfaces at-1/0/0 unit 3]
user@host# set family inet
```

10. Configure the VCI value.

```
[edit interfaces at-1/0/0 unit 3]
user@host# set vci 35
```

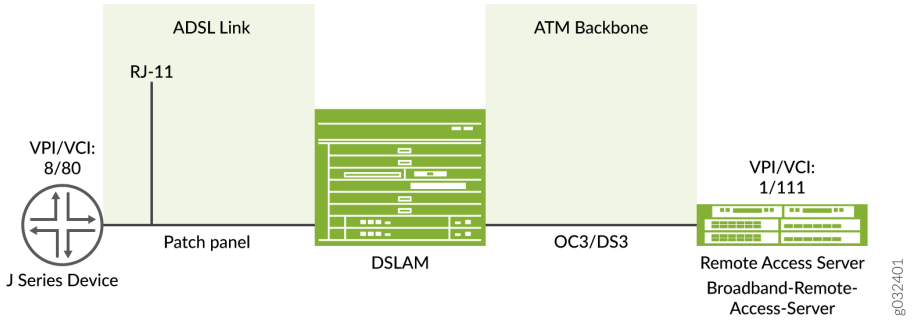
Use the `show` command to see the output of the configuration.

Configure MLPPP-over-ADSL Interfaces

In this example, you set the encapsulation as atm-mlppp-llc for the interface `at-5/0/0`. You then configure the family MLPPP bundle as `lsq-0/0/0.1`.

Figure 4 on page 59 shows a typical example of MLPPP-over-ADSL end-to-end connectivity.

Figure 4: MLPPP-over-ADSL Interface



To configure MLPPP on an ADSL interface:

1. Configure an interface.

```
[edit]
user@host# edit interfaces at-5/0/0 unit 0
```

2. Set the MLPPP encapsulation.

```
[edit interfaces at-5/0/0 unit 0]
user@host# set encapsulation atm-mlppp-llc
```

3. Specify the family MLPPP.

```
[edit interfaces at-5/0/0 unit 0]
user@host# set family mlppp bundle lsq-0/0/0.1
```

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

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

Use the `show` command to see the output of the configuration.

Configure CHAP on DSL Interfaces

In this example, you specify the CHAP access profile and create an interface called `at-3/0/0`. You configure CHAP on either the ATM-over-ADSL or the ATM-over-SHDSL interface and specify a unique profile name called A-ppp-client containing a client list and access parameters. You then specify a unique hostname called `A-at-3/0/0.0` to be used in CHAP. Finally, you set the passive option to handle incoming CHAP packets. To configure CHAP on either the ATM-over-ADSL or the ATM-over-SHDSL interface:

1. Define a CHAP access profile.

```
[edit]
user@host# set access profile A-ppp-client client client1 chap-secret my-secret
```

2. Create an interface.

```
[edit]
user@host# edit interfaces at-3/0/0 unit 0
```

3. Configure CHAP and specify a unique profile name.

```
[edit interfaces at-3/0/0 unit 0]
user@host# set ppp-options chap access-profile A-ppp-client
```

4. Specify a unique hostname.

```
[edit interfaces at-3/0/0 unit 0]
user@host# set ppp-options chap local-name A-at-3/0/0.0
```

5. Set the option to handle incoming CHAP packets only.

```
[edit interfaces at-3/0/0 unit 0]
user@host# set ppp-options chap passive
```

Use the `show` command to see the output of the configuration.

Configure ATM-over-SHDSL Network Interfaces

In this example, you set the ATM-over-SHDSL mode on the G.SHDSL interface, if required. You create an interface called `at-2/0/0` and configure the physical properties for the interface. You configure the encapsulation type and annex type. You specify the SHDSL line rate for the ATM-over-SHDSL interface

and the loopback address for testing the SHDSL connection integrity. Then you configure the SNR margin, set the logical interface, and configure the encapsulation for the ATM-over-SHDSL logical unit.

Additionally, you configure the OAM liveness values for an ATM virtual circuit and set the OAM period, Finally, you add the family protocol type inet and configure the VCI value. To configure ATM-over-SHDSL network interfaces for the device:

1. Set the ATM-over-SHDSL mode on the G.SHDSL interface.

```
[edit]
user@host# set chassis fpc 6 pic 0 shdsl pic-mode 1-port-atm
```

2. Create an interface.

```
[edit]
user@host# edit interfaces at-2/0/0
```

3. Configure the physical properties for the interface.

```
[edit interfaces at-2/0/0]
user@host# set atm-options vpi 25
user@host# set atm-options vpi 25 oam-liveness up-count 200 down-count 200
user@host# set atm-options vpi 25 oam-period 100
```

4. Configure the encapsulation type.

```
[edit interfaces at-2/0/0]
user@host# set encapsulation ethernet-over-atm
```

5. Set the annex type.

```
[edit]
user@host# edit interfaces at-2/0/0 shdsl-options
user@host# set annex annex-a
```

6. Configure the SHDSL line rate.

```
[edit interfaces at-2/0/0 shdsl-options]
user@host# set line-rate auto
```

7. Configure the loopback option for testing the SHDSL connection integrity.

```
[edit interfaces at-2/0/0 shdsl-options]
user@host# set loopback local
```

8. Configure the signal-to-noise ration margin.

```
[edit interfaces at-2/0/0 shdsl-options]
user@host# set snr-margin current 5
user@host# set snr-margin snext5
```

9. Configure the logical interface.

```
[edit]
user@host# edit interfaces at-2/0/0 unit 3
```

10. Configure the encapsulation for the logical unit.

```
[edit interfaces at-2/0/0 unit 3]
user@host# set encapsulation atm-nlpid
```

11. Configure the OAM liveness values for an ATM virtual circuit

```
[edit interfaces at-2/0/0 unit 3]
user@host# set oam-liveness up-count 200 down-count 200
```

12. Configure the OAM period.

```
[edit interfaces at-2/0/0 unit 3]
user@host# set oam-period 100
```

13. Add the Family protocol type.

```
[edit interfaces at-2/0/0 unit 3]
user@host# set family inet
```

14. Configure the VCI value.

```
[edit interfaces at-2/0/0 unit 3]
user@host# set vci 35
```

Use the `show` command to see the output of the configuration.

Verification

Display information about the parameters configured on the ADSL and SHDSL interfaces.

- To verify that the DHCP options are configured use the `run show system services dhcp client` command:

```
user@host# run show system services dhcp client

Logical Interface name         at-1/0/0.0
        Hardware address        00:1f:12:e4:71:38
        Client status           bound
        Address obtained        10.40.1.2
        Update server           disabled
        Lease obtained at       2011-05-03 04:58:10 PDT
        Lease expires at        2011-05-04 04:58:10 PDT

DHCP options:
    Name: server-identifier, Value: 10.40.1.1
    Code: 1, Type: ip-address, Value: 255.255.255.0
    Name: name-server, Value: [ 192.168.5.68, 192.168.60.131, 172.17.28.100, 172.17.28.101 ]
    Name: domain-name, Value: englab.juniper.net
```

To verify the interface status and check traffic statistics use the `show interface terse` command and test end-to-end data path connectivity by sending the ping packets to the remote end IP address:

```
user@host# run show interfaces at-1/0/0 terse

Interface               Admin Link Proto    Local                 Remote
```

```
at-1/0/0                up    up
at-1/0/0.0              up    up   inet     10.40.1.2/24
at-1/0/0.32767          up    up

user@host# run ping 10.40.1.1 count 100 rapid

PING 10.40.1.1 (10.40.1.1): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!
--- 10.40.1.1 ping statistics ---
100 packets transmitted, 100 packets received, 0% packet loss
round-trip min/avg/max/stddev = 20.086/26.404/61.723/6.194 ms
```

- To verify that the ADSL interface properties are configured use the `show ipv6 neighbors` command. The output shows a summary of interface information.

```
user@host> show ipv6 neighbors
IPv6 Address     Linklayer Address      State       Exp Rtr Secure     Interface
            10:1::2          00:00:0a:00:00:00      reachable   17   yes     no
reth0.0
            13:13::1         00:19:e2:4b:61:83      stale       1197 yes     no
at-1/0/0.0
            12:12::2         00:19:e2:4b:61:83      stale       1188 yes     no
at-3/0/0.0
```

The `IPv6 Address` field displays the configured IPv6 address on the interface.

- To verify the ADSL interface properties, use the `show interfaces at-1/0/0 extensive` command:

```
user@host> show interfaces at-1/0/0 extensive
 Physical interface: at-1/0/0, Enabled, Physical link is Up
   Interface index: 141, SNMP ifIndex: 49, Generation: 142
   Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, ADSL mode,
   Speed: ADSL, Loopback: None
   Device flags   : Present Running
   Link flags     : None
   CoS queues     : 8 supported, 8 maximum usable queues
   Hold-times     : Up 0 ms, Down 0 ms
   Current address: 00:05:85:c3:17:f4
   Last flapped   : 2008-06-26 23:11:09 PDT (01:41:30 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, Invalid VCs: 0, Framing errors: 0,Policed discards: 0, L3
incompletes: 0, L2 channelerrors: 0, L2 mismatch timeouts: 0,
     Resource errors: 0
   Output errors:
     Carrier transitions: 3, Errors: 0, Drops: 0, Aged packets: 0, MTU errors: 0, Resource
errors: 0
   ADSL  alarms   : None
   ADSL  defects  : None
   ADSL media:            Seconds        Count State
    LOF                       1            1  OK
    LOS                       1            1  OK
    LOM                       0            0  OK
    LOP                       0            0  OK
    LOCDI                     0            0  OK
    LOCDNI                    0            0  OK
   ADSL status:
     Modem status  : Showtime (Adsl2plus)
     DSL mode      :     Auto    Annex A
     Last fail code: None
     Subfunction   : 0x00
     Seconds in showtime : 6093
   ADSL Chipset Information:              ATU-R               ATU-C
     Vendor Country :                      0x0f                0xb5
     Vendor ID      :                      STMI               IFTN
     Vendor Specific:                      0x0000             0x70de
   ADSL Statistics:                       ATU-R               ATU-C
     Attenuation (dB)         :            0.0                 0.0
     Capacity used(%)         :            100                 92
     Noise margin(dB)         :            7.5                 9.0
     Output power (dBm)       :            10.0                12.5
                                  Interleave       Fast  Interleave         Fast
     Bit rate (kbps)   :                  0      24465           0         1016
     CRC               :                  0          0           0         0
     FEC               :                  0          0           0         0
     HEC               :                  0          0           0         0
     Received cells    :                  0         49
     Transmitted cells :                  0          0
```

```
   ATM status:
     HCS state:     Hunt
     LOC      :       OK
  ATM Statistics:
    Uncorrectable HCS errors: 0, Correctable HCS errors: 0,Tx cell FIFO overruns: 0,Rx cell
FIFO overruns: 0,Rx cell FIFO underruns: 0,
    Input cell count: 49, Output cell count: 0,Output idle cell count: 0,Output VC queue
drops: 0Input no buffers: 0, Input length errors: 0,
    Input timeouts: 0, Input invalid VCs: 0, Input bad CRCs: 0, Input OAM cell no buffers: 0

  Packet Forwarding Engine configuration:
     Destination slot: 1
     Direction : Output
     CoS transmit queue               Bandwidth              Buffer  Priority   Limit
                              %            bps     %           usec
     0 best-effort            95        7600000    95              0       low
none
     3 network-control         5         400000     5              0       low    none



     But for ADSL MiniPim TI chipset does not send  ADSL Chipset
     Information. Also Adsl minipim does not send any alarms. So we can't
     show alarm stats for minipim. So following information will not be
     displayed in Minipim case.

     ADSL  alarms   : None
     ADSL  defects  : None
     ADSL media:            Seconds        Count State
     LOF                       1            1  OK
     LOS                       1            1  OK
     LOM                       0            0  OK
     LOP                       0            0  OK
     LOCDI                     0            0  OK
     LOCDNI                    0            0  OK

    ADSL Chipset Information:              ATU-R      ATU-C
      Vendor Country :                      0x0f       0xb5
      Vendor ID      :                      STMI       IFTN
      Vendor Specific:                      0x0000     0x70de
```

The output shows a summary of interface information.

To verify the PPPoA configuration for an ATM-over-ADSL interface is correct, use the the `show interfaces at-1/0/0` and the `show access` commands.

- To verify the configuration for an MLPPP-over-ADSL Interface is correct, use the `show interfaces at-5/0/0` command.
- To verify that the ADSL interface properties are enabled, use the `show interfaces at-3/0/0 extensive` command.

```
user@host> show interfaces at-3/0/0 extensive
 Physical interface: at-3/0/0, Enabled, Physical link is Up
   Interface index: 141, SNMP ifIndex: 49, Generation: 142
   Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, ADSL mode,
   Speed: ADSL, Loopback: None
   Device flags   : Present Running
   Link flags     : None
   CoS queues     : 8 supported, 8 maximum usable queues
   Hold-times     : Up 0 ms, Down 0 ms
   Current address: 00:05:85:c3:17:f4
   Last flapped   : 2008-06-26 23:11:09 PDT (01:41:30 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, Invalid VCs: 0, Framing errors: 0,Policed discards: 0, L3
incompletes: 0, L2 channelerrors: 0, L2 mismatch timeouts: 0,
     Resource errors: 0
   Output errors:
     Carrier transitions: 3, Errors: 0, Drops: 0, Aged packets: 0, MTU errors: 0, Resource
errors: 0
   ADSL  alarms   : None
   ADSL  defects  : None
   ADSL media:            Seconds        Count State
    LOF                       1            1  OK
    LOS                       1            1  OK
    LOM                       0            0  OK
    LOP                       0            0  OK
    LOCDI                     0            0  OK
    LOCDNI                    0            0  OK
   ADSL status:
```

```
     Modem status  : Showtime (Adsl2plus)
     DSL mode      :     Auto    Annex A
     Last fail code: None
     Subfunction   : 0x00
     Seconds in showtime : 6093
   ADSL Chipset Information:              ATU-R               ATU-C
     Vendor Country :                      0x0f                0xb5
     Vendor ID      :                      STMI               IFTN
     Vendor Specific:                      0x0000             0x70de
   ADSL Statistics:                       ATU-R               ATU-C
     Attenuation (dB)         :            0.0                 0.0
     Capacity used(%)         :            100                 92
     Noise margin(dB)         :            7.5                 9.0
     Output power (dBm)       :            10.0                12.5
                                  Interleave       Fast  Interleave         Fast
     Bit rate (kbps)   :                  0      24465           0         1016
     CRC               :                  0          0           0         0
     FEC               :                  0          0           0         0
     HEC               :                  0          0           0         0
     Received cells    :                  0         49
     Transmitted cells :                  0          0
   ATM status:
     HCS state:     Hunt
     LOC      :       OK
  ATM Statistics:
    Uncorrectable HCS errors: 0, Correctable HCS errors: 0,Tx cell FIFO overruns: 0,Rx cell
FIFO overruns: 0,Rx cell FIFO underruns: 0,
    Input cell count: 49, Output cell count: 0,Output idle cell count: 0,Output VC queue
drops: 0Input no buffers: 0, Input length errors: 0,
    Input timeouts: 0, Input invalid VCs: 0, Input bad CRCs: 0, Input OAM cell no buffers: 0

  Packet Forwarding Engine configuration:
     Destination slot: 1
     Direction : Output
     CoS transmit queue               Bandwidth              Buffer  Priority   Limit
                              %            bps     %           usec
     0 best-effort            95        7600000    95              0       low
none
     3 network-control         5         400000     5              0       low    none



     But for ADSL MiniPim TI chipset does not send  ADSL Chipset
```

```
    Information. Also Adsl minipim does not send any alarms. So we can't
    show alarm stats for minipim. So following information will not be
    displayed in Minipim case.

    ADSL  alarms   : None
    ADSL  defects  : None
    ADSL media:            Seconds        Count State
    LOF                       1            1  OK
    LOS                       1            1  OK
    LOM                       0            0  OK
    LOP                       0            0  OK
    LOCDI                     0            0  OK
    LOCDNI                    0            0  OK

   ADSL Chipset Information:               ATU-R      ATU-C
     Vendor Country :                      0x0f       0xb5
     Vendor ID      :                      STMI       IFTN
     Vendor Specific:                      0x0000     0x70de
```

To verify the PPPoA configuration for an ATM-over-ADSL interface is correct, use the `show interfaces at-3/0/0` and the `show access` commands.

To verify that an ATM-over-SHDSL configuration is correct, use the `show interfaces at-3/0/0 extensive` command.

```
user@host> show interfaces at-3/0/0 extensive
Physical interface: at-3/0/0, Enabled, Physical link is Up
  Interface index: 141, SNMP ifIndex: 23, Generation: 48
  Link-level type: ATM-PVC, MTU: 4482, Clocking: Internal, ADSL mode, Speed: ADSL,
  Loopback: None
  Device flags   : Present Running
  Link flags     : None
  CoS queues     : 8 supported
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:05:85:c7:44:3c
  Last flapped   : 2005-05-16 05:54:41 PDT (00:41:42 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                 4520                    0 bps
   Output bytes  :                39250                    0 bps
   Input  packets:                   71                    0 pps
   Output packets:                 1309                    0 pps
```

```
Input errors:
  Errors: 0, Drops: 0, Invalid VCs: 0, Framing errors: 0, Policed discards: 0,
  L3 incompletes: 0, L2 channel errors: 1, L2 mismatch timeouts: 0, Resource errors: 0
Output errors:
  Carrier transitions: 3, Errors: 0, Drops: 0, Aged packets: 0, MTU errors: 0,
  Resource errors: 0
Queue counters:       Queued packets  Transmitted packets      Dropped packets
  0 best-effort                    4                    4                    0
  1 expedited-fo                   0                    0                    0
  2 assured-forw                   0                    0                    0
  3 network-cont                2340                 2340                    0
SHDSL  alarms   : None
SHDSL  defects  : None
SHDSL media:            Seconds        Count  State
  LOSD                  239206            2  OK
  LOSW                  239208            1  OK
  ES                         3            1  OK
  SES                        0            0  OK
  UAS                        3            1  OK

SHDSL status:
  Line termination     :STU-R
   Annex                    :Annex B
   Line Mode                :2-wire
   Modem Status            :Data
   Last fail code        :0
   Framer mode            :ATM
   Dying Gasp               :Enabled
   Chipset version        :1
   Firmware version        :R3.0
SHDSL Statistics:
 Loop Attenuation (dB)           :0.600
  Transmit power (dB)                :8.5
  Receiver gain (dB)               :21.420
  SNR sampling (dB)                  :39.3690
  Bit rate     (kbps)                  :2304
  Bit error rate                      :0
  CRC errors                          :0
  SEGA errors                          :1
  LOSW errors                          :0
  Received cells                      :1155429
  Transmitted cells                  :1891375
```

```
    HEC errors                          :0
    Cell drop                          :0
```

Example: Configure G.SHDSL Interface

This example shows how to configure the G.SHDSL interface on SRX Series Firewalls.

To configure GSHDSL interface:

1. Specify the wire mode on the G.SHDSL interface. The default wire mode is 4-wire (2-port, 4-wire).
2. Specify the annex type. The default annex type is `auto`.
3. Specify the SHDSL line rate (speed of transmission of data on the SHDSL connection). The default line rate is `auto`.
4. Specify the encapsulation type. The `pt-` interface does not require encapsulation types.
5. Configure the encapsulation type.

Before you begin:

- Configure the network interfaces as necessary. See "Understanding Ethernet Interfaces" on page 125.
- Install the G.SHDSL Mini-PIM in the first slot of the SRX210 chassis.
- Connect the SRX210 device to a DSLAM (IP DSLAM and ATM DSLAM).

Figure 5 on page 72 shows the topology for the G.SHDSL Mini-PIM operating in 2X4-wire mode.

Figure 5: G.SHDSL Mini-PIM Operating in 2X4-Wire Mode

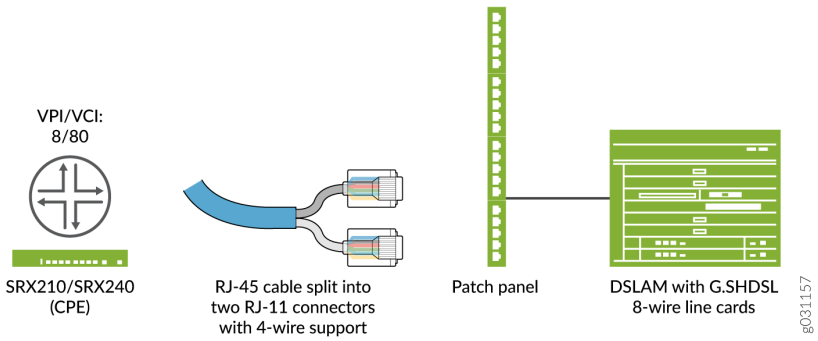


Figure 6 on page 72 shows the topology for the G.SHDSL Mini-PIM operating in 4X2-wire mode.

Figure 6: G.SHDSL Mini-PIM Operating in 4X2-Wire Mode

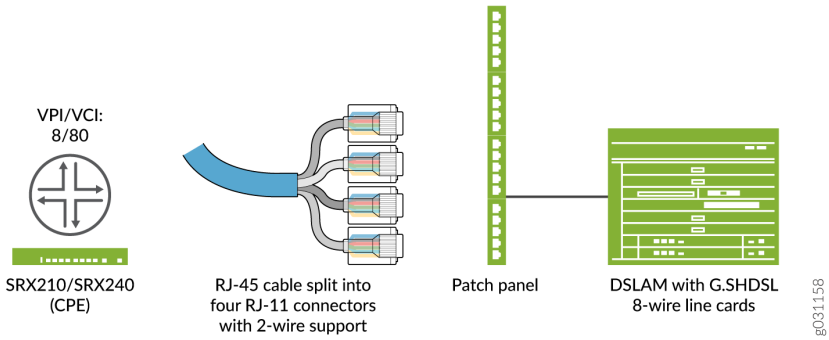
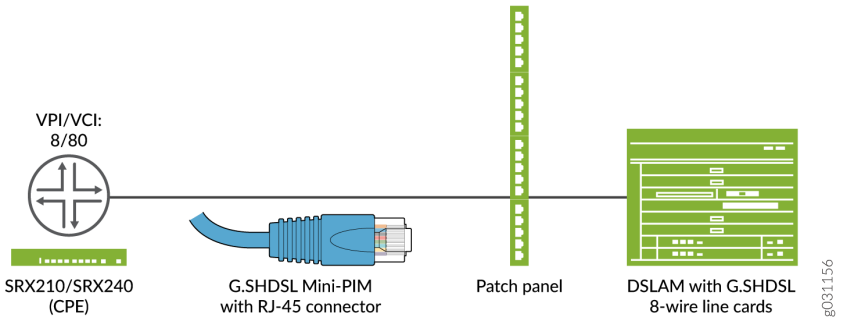


Figure 7 on page 73 shows the topology for the G.SHDSL Mini-PIM operating in 1X8-wire mode.

Figure 7: G.SHDSL Mini-PIM Operating in 1X8-Wire Mode



Determine the operating wire mode (2-wire, 4-wire, or 8-wire) and corresponding CLI code listed in Table 15 on page 73.

Table 15: Operating Wire Modes

Wire Mode Configuration	CLI Code
2x4-wire Configuration	`set chassis fpc 1 pic 0 shdsl pic-mode 2-port-atm` **NOTE**: The 2x4-wire configuration is the default configuration and behavior.
4x2-wire Configuration	`set chassis fpc 1 pic 0 shdsl pic-mode 4-port-atm`
1x8-wire Configuration	`set chassis fpc 1 pic 0 shdsl pic-mode 1-port-atm`

When you set the wire mode to 8-wire, one physical interface (IFD) is created. Similarly for 4-wire mode and 2-wire mode, two IFDs and four IFDs are created, respectively.

In this example:

1. First configure a basic G.SHDSL interface. Set the operation wire mode to 2-port-atm, the line rate to 4096, and the annex type to `annex-a`.
2. Configure the G.SHDSL interface when the device is connected to an IP DSLAM. Set the type of encapsulation to ethernet-over-atm and the ATM VPI option to 0. Set the type of encapsulation on the G.SHDSL logical interface as `ether-over-atm-llc` and configure the ATM VCI option to 0.60. Also, set the interface address for the logical interface to 10.1.1.1/24.

3. Configure the G.SHDSL interface when the device is connected to an ATM DSLAM. Set the ATM VPI to 0 and set the type of encapsulation to `ppp-over-ether-over-atm-llc`. Specify a PPPoE interface with the PAP access profile, `local-name`, and `local-password`. Configure the passive option to handle incoming PAP packets and set the logical interface as the underlying interface for the PPPoE session to `at-1/0/0.0`. Set the number of seconds to 120 to wait before reconnecting after a PPPoE session is terminated. Specify the logical interface as the client for the PPPoE interface and obtain an IP address by negotiation with the remote end.

4. Configure PPPoA over ATM for the G.SHDSL Interface. set the type of encapsulation to `atm-pvc` and the ATM VPI to 0. Set the type of encapsulation for PPP over ATM adaptation layer 5 (AAL5) logical link control (LLC) on the logical interface and set the ATM VCI to 122. Configure the PPPoA interface with the CHAP access profile as juniper and set the `local-name` for the CHAP interface to srx-210. Finally, you obtain an IP address by negotiation with the remote end.

Table 16 on page 74specifies the CLI quick configuration commands used for configuring GSHDSL interfaces.

Table 16: CLI Quick Configuration

Configuration Step	CLI Quick Configuration Commands
Configure a basic G.SHDSL interface	**set chassis fpc 1 pic 0 shdsl pic-mode 2-port-atm** **set interfaces at-1/0/0 shdsl-options line-rate 4096 annex annex-a**
Configure G.SHDSL interface when connected to an IP DSLAM	**set interfaces at-1/0/0 encapsulation ethernet-over-atm** **set interfaces at-1/0/0 atm-options vpi 0** **set interfaces at-1/0/0 unit 0 encapsulation ether-over-atm-llc vci 0.60** **set interfaces at-1/0/0 unit 0 family inet address 10.1.1.1/24**
Configure G.SHDSL Interface when connected to an ATM DSLAM	**set interfaces at-1/0/0 encapsulation atm-pvc atm-options vpi 0** **set interfaces at-1/0/0 unit 0 encapsulation atm-snap vci 0.65** **set interfaces at-1/0/0 unit 0 family inet address 10.2.1.1/24**

Table 16: CLI Quick Configuration *(Continued)*

Configuration Step	CLI Quick Configuration Commands

Configure PPPoE over ATM for the G.SHDSL interface	`set interfaces at-1/0/0 encapsulation ethernet-over-atm  atm-options vpi 0`
	`set interfaces at-1/0/0 unit 0 encapsulation ppp-over-ether-over-atm-llc vci 0.35`
	`set interfaces pp0 unit 0 ppp-options pap access-profile pap_prof local-name srx-210`
	`set interfaces pp0 unit 0 ppp-options pap local-password  "$9$0tLw1SeN-woJDSr-wY2GU69Cp1RSre"`
	`set interfaces pp0 unit 0 ppp-options pap passive`
	`set interfaces pp0 unit 0 pppoe-options underlying-interface at-1/0/0.0`
	`set interfaces pp0 unit 0 pppoe-options auto-reconnect 120 client`
	`set interfaces pp0 unit 0 family inet negotiate-address`

Configure PPPoA over ATM for the G.SHDSL interface	`set interfaces at-1/0/0 encapsulation atm-pvc atm-options vpi 0`
	`set interfaces at-1/0/0 unit 0 encapsulation atm-ppp-llc vci 1.122`
	`set interfaces at-1/0/0 unit 0 ppp-options chap access-profile juniper local-name srx-210`
	`set interfaces at-1/0/0 unit 0 family inet negotiate-address`

Cofigure a basic G.SHDSL interface in EFM PIC mode	`set chassis fpc 1 pic 0 shdsl pic-mode efm`
	`set interfaces pt-1/0/0 shdsl-options annex annex-g`
	`set interfaces pt-1/0/0 shdsl-options line-rate 5696`
	`set interfaces pt-1/0/0 unit 0 family inet address 10.10.10.1/24`

Configure PPPoE and VLAN for the G.SHDSL EFM interface	`set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether`
	`set interfaces pp0 unit 0 ppp-options pap access-profile pap_prof local-name srx-210`
	`set interfaces pp0 unit 0 ppp-options pap local-password  "$9$0tLw1SeN-woJDSr-wY2GU69Cp1RSre"`
	`set interfaces pp0 unit 0 ppp-options pap passive`
	`set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0`
	`set interfaces pp0 unit 0 pppoe-options auto-reconnect 120 client`
	`set interfaces pp0 unit 0 family inet negotiate-address`

Table 16: CLI Quick Configuration *(Continued)*

Configuration Step	CLI Quick Configuration Commands

Configure a Basic G.SHDSL Interface in EFM PIC Mode	`set chassis fpc 1 pic 0 shdsl pic-mode efm`
	`set interfaces pt-1/0/0 shdsl-options annex annex-g`
	`set interfaces pt-1/0/0 shdsl-options line-rate 5696`
	`set interfaces pt-1/0/0 unit 0 family inet address 10.10.10.1/24`

Configure PPPoE and VLAN for the G.SHDSL EFM Interface	`set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether`
	`set interfaces pp0 unit 0 ppp-options pap access-profile pap_prof local-name srx-210`
	`set interfaces pp0 unit 0 ppp-options pap local-password "$9$0tLw1SeN-woJDSr-wY2GU69Cp1RSre"`
	`set interfaces pp0 unit 0 ppp-options pap passive`
	`set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0`
	`set interfaces pp0 unit 0 pppoe-options auto-reconnect 120 client`
	`set interfaces pp0 unit 0 family inet negotiate-address`

Configure the Basic G.SHDSL Interfaces

To view the CLI quick configuration commands, see Table 16 on page 74. To configure the basic G.SHDSL interface on SRX210 devices:

1. Select the operating wire mode.

```
[edit]
user@host# set chassis fpc 1 pic 0 shdsl pic-mode 2-port-atm
```

2. Create an interface and set options.

```
[edit]
user@host# edit interfaces at-1/0/0 shdsl-options
```

3. Configure the line rates.

```
[edit interfaces at-1/0/0 shdsl-options]
user@host# set line-rate 4096
```

4. Set the annex type.

```
[edit interfaces at-1/0/0 shdsl-options]
user@host# set annex annex-a
```

Use the `show interfaces at-1/0/0` command to see the output of the configuration.

Configure a G.SHDSL Interface When Connected to an IP DSLAM

To configure the G.SHDSL interface on an SRX210 device when the device is connected to an IP DSLAM: :

1. Create an interface.

```
[edit]
user@host# edit interfaces at-1/0/0
```

2. Specify the type of encapsulation.

```
[edit interfaces at-1/0/0]
user@host# set encapsulation ethernet-over-atm
```

3. Configure the ATM VPI option.

```
[edit interfaces at-1/0/0]
user@host# set atm-options vpi 0
```

4. Specify the type of encapsulation for logical interface.

```
[edit interfaces at-1/0/0 ]
user@host# edit unit 0
user@host# set encapsulation ether-over-atm-llc
```

5. Configure the ATM VCI options for the logical interface.

```
[edit interfaces at-1/0/0 unit 0]
user@host# set vci 0.60
```

6. Configure the interface address.

```
[edit interfaces at-1/0/0 unit 0]
user@host# set family inet address 10.1.1.1/24
```

Use the `show interfaces at-1/0/0` command to see the output of the configuration.

Configure a G.SHDSL Interface When Connected to an ATM DSLAM

To configure the G.SHDSL interface on an SRX210 device when the device is connected to an ATM DSLAM: :

1. Create an interface.

```
[edit]
user@host# edit interfaces at-1/0/0
```

2. Specify the type of encapsulation.

```
[edit interfaces at-1/0/0]
user@host# set encapsulation atm-pvc
```

3. Configure the ATM VPI option.

```
[edit interfaces at-1/0/0]
user@host# set atm-options vpi 0
```

4. Specify the type of encapsulation for the logical interface.

```
[edit interfaces at-1/0/0]
user@host# edit unit 0
user@host# set encapsulation atm-snap
```

5. Configure the ATM VCI option.

```
[edit interfaces at-1/0/0 unit 0]
user@host# set vci 0.65
```

6. Configure the interface address.

```
[edit interfaces at-1/0/0 unit 0]
user@host# set family inet address 10.2.1.1/24
```

Use the `show interfaces at-1/0/0` command to see the output of the configuration.

Configure PPPoE over ATM for the G.SHDSL Interface

To configure PPPoE over ATM on the G.SHDSL interface:

1. Create an interface.

```
[edit]
user@host# edit interfaces at-1/0/0
```

2. Specify the type of encapsulation.

```
[edit interfaces at-1/0/0]
user@host# set encapsulation ethernet-over-atm
```

3. Configure the ATM VPI option.

```
[edit interfaces at-1/0/0]
user@host# set atm-options vpi 0
```

4. Specify the type of encapsulation on the logical interface.

```
[edit interfaces at-1/0/0]
user@host# edit unit 0
user@host# set encapsulation ppp-over-ether-over-atm-llc
```

5. Configure the ATM VCI option.

```
[edit interfaces at-1/0/0 unit 0]
user@host# set vci 0.35
```

6. Configure a PPPoE interface with the PAP access profile.

```
[edit]
user@host# edit interfaces pp0 unit 0 ppp-options pap
user@host# set access-profile pap_prof
```

7. Configure a local-name for the PAP interface.

```
[edit interfaces pp0 unit 0 ppp-options pap]
user@host# set local-name srx-210
```

8. Configure a local-password for the PAP interface.

```
[edit interfaces pp0 unit 0 ppp-options pap]
user@host# set local-password "$9$0tLw1SeN-woJDSr-wY2GU69Cp1RSre"
```

9. Set the passive option to handle incoming PAP packets.

```
[edit interfaces pp0 unit 0 ppp-options pap]
user@host# set passive
```

10. Specify the logical interface as the underlying interface for the PPPoE session.

```
[edit]
user@host# edit interfaces pp0 unit 0 pppoe-options
user@host# set underlying-interface at-1/0/0.0
```

11. Specify the number of seconds.

```
[edit interfaces pp0 unit 0 pppoe-options]
user@host# set auto-reconnect 120
```

12. Set the logical interface as the client for the PPPoE interface.

```
[edit interfaces pp0 unit 0 pppoe-options]
user@host# set client
```

13. Obtain an IP address by negotiation with the remote end.

```
[edit]
user@host# edit interfaces pp0 unit 0
user@host# set family inet negotiate-address
```

Use the `show interfaces at-1/0/0` and `show interfaces pp0` commands to see the output of the configuration.

Configure PPPoA over ATM for the G.SHDSL Interface

To configure PPPoA over ATM on the G.SHDSL interface:

1. Create an interface.

```
[edit]
user@host# edit interfaces at-1/0/0
```

2. Specify the type of encapsulation.

```
[edit interfaces at-1/0/0]
user@host# set encapsulation atm-pvc
```

3. Configure the ATM VPI option.

```
[edit interfaces at-1/0/0]
user@host# set atm-options vpi 0
```

4. Specify the type of encapsulation on the G.SHDSL logical interface.

```
[edit]
user@host# edit interfaces at-1/0/0 unit 0
user@host# set encapsulation atm-ppp-llc
```

5. Configure the ATM VCI option.

```
[edit interfaces at-1/0/0 unit 0]
user@host# set vci 1.122
```

6. Configure a PPPoA interface with the CHAP access profile.

```
[edit]
user@host# edit interfaces at-1/0/0 unit 0 ppp-options chap
user@host# set access-profile juniper
```

7. Configure a local name for the CHAP interface.

```
[edit interfaces at-1/0/0 unit 0 ppp-options chap]
user@host# set local-name srx-210
```

8. Obtain an IP address by negotiation with the remote end.

```
[edit]
user@host# edit interfaces at-1/0/0 unit 0
user@host# set family inet negotiate-address
```

Use the `show interfaces at-1/0/0` command to see the output of the configuration.

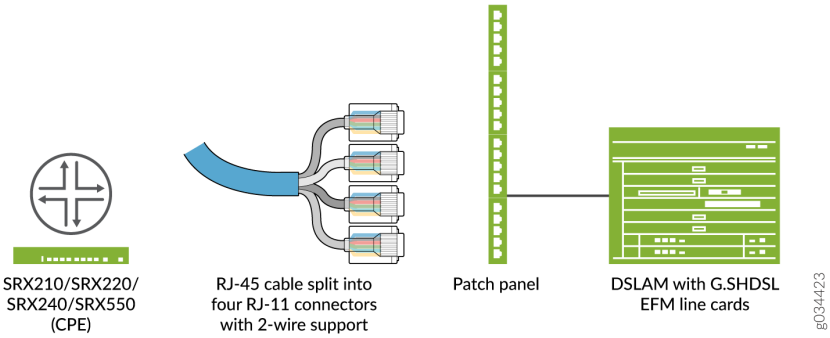
Configure G.SHDSL Interface in EFM Mode

In this example:

1. You first configure a basic G.SHDSL interface by setting the operation wire mode to efm, the line rate to auto, and the annex type to annex-auto.
2. You then configure the G.SHDSL interface when the device is connected to an EFM IP DSLAM. You set the logical interface to 10.10.10.1/24.
3. Next you configure PPPoE for the G.SHDSL Interface. Configure the encapsulation as ppp-over-ether under `unit 0` of `pt-1/0/0` interface. You specify a PPPoE interface with the PAP access profile, local name, and local password. Then you configure the passive option to handle incoming PAP packets and set the logical interface as the underlying interface for the PPPoE session to `pt-1/0/0.0`. Also, you set the number of seconds to 120 to wait before reconnecting after a PPPoE session is terminated. (The range is 1 through 4,294,967,295 seconds.) Finally, you specify the logical interface as the client for the PPPoE interface and obtain an IP address by negotiation with the remote end.

Figure 8 on page 83 shows the topology for the G.SHDSL Mini-PIM operating in EFM mode.

Figure 8: G.SHDSL Mini-PIM Operating in EFM Mode



For operating wire mode EFM configuration, use the `set chassis fpc 1 pic 0 shdsl pic-mode efm` CLI code. When PIC mode is set to EFM, an interface called `pt-1/0/0` is created.

To view the CLI quick configuration commands, see Table 16 on page 74.

Configure a Basic G.SHDSL Interface in EFM PIC Mode

To configure a basic G.SHDSL interface:

1. Specify the PIC mode.

```
[edit]
user@host# set chassis fpc 1 pic 0 shdsl pic-mode efm
```

> **NOTE**: When configuring the G.SHDSL interface in chassis cluster mode, include the node ID. For example, to configure the G.SHDSL interface (operating in EFM PIC mode) in chassis cluster mode for fpc slot 1 on node 0, use the following command:
>
> **set chassis node 0 fpc 1 pic 0 shdsl pic-mode efm**

2. Configure the IP address.

```
[edit]
user@host# set interfaces pt-1/0/0 unit 0 family inet address 10.10.10.1/24
```

NOTE: By default, annex mode and line rate are set to auto. If you have to configure annex mode (annex-g) and line rate (5696 Kbps), follow Steps 3, 4, and 5.

3. Configure SHDSL options.

```
[edit]
user@host# set interfaces pt-1/0/0 shdsl-options
```

4. Specify the annex type.

```
[edit interfaces pt-1/0/0 shdsl-options]
user@host# set annex annex-g
```

5. Configure the line rate.

```
[edit interfaces pt-1/0/0 shdsl-options]
user@host# set line-rate 5696
```

Use the `show interfaces pt-1/0/0` and `show chassis fpc 1` commands to see the output of the configuration.

Configure a PPPoE and VLAN for the G.SHDSL EFM Interface

To configure PPPoE and VLAN for the G.SHDSL EFM Interface:

1. Create an interface.

```
[edit]
user@host# set interfaces pt-1/0/0
```

2. Specify the type of encapsulation.

```
[edit interfaces pt-1/0/0]
user@host# set unit 0
user@host# set encapsulation ppp-over-ether
```

3. Configure a PPPoE interface with the PAP access profile.

```
[edit]
user@host# set interfaces pp0 unit 0 ppp-options pap
user@host# set access-profile pap_prof
```

4. Configure a local name for the PAP interface.

```
[edit interfaces pp0 unit 0 ppp-options pap]
user@host# set local-name srx-210
```

5. Configure a local password for the PAP interface.

```
[edit interfaces pp0 unit 0 ppp-options pap]
user@host# set local-password "$9$0tLw1SeN-woJDSr-wY2GU69Cp1RSre"
```

6. Set the passive option to handle incoming PAP packets.

```
[edit interfaces pp0 unit 0 ppp-options pap]
user@host# set passive
```

7. Specify the logical interface as the underlying interface for the PPPoE session.

```
[edit]
user@host# set interfaces pp0 unit 0 pppoe-options
user@host# set underlying-interface pt-1/0/0.0
```

8. Specify the number of seconds.

```
[edit interfaces pp0 unit 0 pppoe-options]
user@host# set auto-reconnect 120
```

9. Set the logical interface as the client for the PPPoE interface.

```
[edit interfaces pp0 unit 0 pppoe-options]
user@host# set client
```

10. Obtain an IP address by negotiation with the remote end.

```
[edit interfaces]
user@host# set pp0 unit 0 family inet negotiate-address
```

11. Configure VLAN on EFM.

```
[edit interfaces]
user@host# set pt-1/0/0 vlan-tagging
```

12. Specify the VLAN ID.

```
[edit interfaces]
user@host# set pt-1/0/0 unit 0 vlan-id 99
```

Use the `show interfaces pt-1/0/0` and `show interfaces pp0` commands to see the output of the configuration.

Verification

Display information about the parameters configured on the GSHDSL interfaces.

- To display information about all the basic G.SHDSL interface properties, use the `show interfaces at-1/0/0 extensive` command.
- To display information about G.SHDSL interface properties:

```
user@host> show interfaces pt-1/0/0 extensive
```

EFM mode for interface pt-1/0/0:

```
Physical interface: pt-1/0/0, Enabled, Physical link is Up
  Interface index: 158, SNMP ifIndex: 575, Generation: 277
  Link-level type: Ethernet, MTU: 1514, Speed: SHDSL(8-Wire)
  Device flags   : Present Running
```

```
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:fe:3d:60:2f:99
  Last flapped   : 2012-10-11 00:03:13 PDT (00:28:57 ago)
  Statistics last cleared: 2012-10-11 00:32:05 PDT (00:00:05 ago)
  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, Invalid VCs: 0, Framing errors: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, Aged packets: 0, MTU errors: 0, Resource
errors: 0
  EFM Group Statistics:
    Type                 : EFM bond
    Active Pairs         : 4
    Bit rate (in Kbps)   : 22784
  Line Pair  0  : Up
    Active alarms  : None
    Active defects : None
    SHDSL media:      Seconds   Count   State
      ES                    0
      SES                   0
      UAS                   0
    SHDSL status:
      Line  termination     : STU-R
      Annex                 : Annex G
      Line mode             : 2-wire
      Modem status          : Data
      Bit rate (kbps)       : 5696
      Last fail mode        : No failure (0x00)
      Framer mode           : EFM
      PAF Status            : Active
      Dying gasp            : Enabled
      Framer sync status    : In sync
    SHDSL statistics:
      Loop attenuation (dB)  : 0.0
      Transmit power (dBm)   : 14.0
      SNR sampling (dB)      : 14.0000
```

```
      CRC errors             : 2
      SEGA errors            : 0
      LOSW errors            : 0
  Line Pair  1  : Up
    Active alarms  : None
    Active defects : None
    SHDSL media:      Seconds   Count   State
      ES                    0
      SES                   0
      UAS                   0
    SHDSL status:
      Line  termination      : STU-R
      Annex                  : Annex G
      Line mode              : 2-wire
      Modem status           : Data
      Bit rate (kbps)        : 5696
      Last fail mode         : No failure (0x00)
      Framer mode            : EFM
      PAF Status             : Active
      Dying gasp             : Enabled
      Framer sync status     : In sync
    SHDSL statistics:
      Loop attenuation (dB)  : 0.0
      Transmit power (dBm)   : 14.0
      SNR sampling (dB)      : 19.0000
      CRC errors             : 0
      SEGA errors            : 0
      LOSW errors            : 0
  Line Pair  2  : Up
    Active alarms  : None
    Active defects : None
    SHDSL media:      Seconds   Count   State
      ES                    0
      SES                   0
      UAS                   0
    SHDSL status:
      Line  termination      : STU-R
      Annex                  : Annex G
      Line mode              : 2-wire
      Modem status           : Data
      Bit rate (kbps)        : 5696
      Last fail mode         : No failure (0x00)
      Framer mode            : EFM
```

```
      PAF Status             : Active
      Dying gasp             : Enabled
      Framer sync status     : In sync
    SHDSL statistics:
      Loop attenuation (dB)  : 0.0
      Transmit power (dBm)   : 14.0
      SNR sampling (dB)      : 14.0000
      CRC errors             : 0
      SEGA errors            : 0
      LOSW errors            : 0
  Line Pair  3  : Up
    Active alarms  : None
    Active defects : None
    SHDSL media:      Seconds   Count   State
      ES                    0
      SES                   0
      UAS                   0
    SHDSL status:
      Line  termination      : STU-R
      Annex                  : Annex G
      Line mode              : 2-wire
      Modem status           : Data
      Bit rate (kbps)        : 5696
      Last fail mode         : No failure (0x00)
      Framer mode            : EFM
      PAF Status             : Active
      Dying gasp             : Enabled
      Framer sync status     : In sync
    SHDSL statistics:
      Loop attenuation (dB)  : 1.0
      Transmit power (dBm)   : 14.0
      SNR sampling (dB)      : 18.0000
      CRC errors             : 0
      SEGA errors            : 0
      LOSW errors            : 0
  Packet Forwarding Engine configuration:
    Destination slot: 0 (0x00)
  CoS information:
    Direction : Output
    CoS transmit queue               Bandwidth               Buffer Priority   Limit
                               %            bps     %           usec
```

```
    0 best-effort            95       21644800    95             0      low    none
    3 network-control         5        1139200     5             0      low    none
```

The output shows a summary of interface information.

RELATED DOCUMENTATION

- Understanding Point-to-Point Protocol over Ethernet | 303
- *Using the CLI Editor in Configuration Mode*
- Configuring the inet6 IPv6 Protocol Family | 36

VDSL2 Interfaces

SUMMARY

Learn about VDSL2 interface details and how to configure the interfaces on security devices.

IN THIS SECTION

VDSL2 Interface Overview

IN THIS SECTION

Very-high-bit-rate digital subscriber line (VDSL) technology is part of the xDSL family of modem technologies, which provide faster data transmission over a single flat untwisted or twisted pair of copper wires. Table 17 on page 91 specifies the key details of the VDSL2 interface.

Table 17: VDSL2 Interface Details

Interface Details	Description
Interface name	SRX-MP-1VDSL2-R
Supported on	For information about platforms support, see hardware compatibility tool (HCT).
Interface type	• `pt-` represents VDSL2 interface when you configure pt- to function as VDSL2. • Interface `pt-1/0/0` comes up by default.
Use cases	• Connects you and the service provider networks over a single connection to provide high bandwidth applications (triple-play services) like high-speed Internet access, Telephone services (VoIP (Voice over IP protocol), High-Definition TV (HDTV)), and Interactive gaming services. • VDSL2 carries the data and multimedia on the copper wire without interrupting the line's ability to carry voice signals. VDSL2 provides an ADSL interface in an ATM DSLAM topology and a VDSL2 interface in an IP or VDSL DSLM topology.

For information on VDSL2 hardware specifications, see 1-Port VDSL2 Annex A Mini-Physical Interface Module (SRX-MP-1VDSL2-R).

Features Supported on the VDSL2 Interface

Table 18 on page 91 describes the key features supported on VDSL2 interface.

Table 18: Key Features Supported on VDSL2

Feature	Description
Packet Transfer Mode (PTM)	• Uses the named interface `pt-1/0/0` and transports packets (IP, PPP, Ethernet, MPLS, and so on) over DSL links as an alternative to using Asynchronous Transfer Mode (ATM). • Based on the Ethernet in the First Mile (EFM) IEEE802.3ah standard.

Table 18: Key Features Supported on VDSL2 *(Continued)*

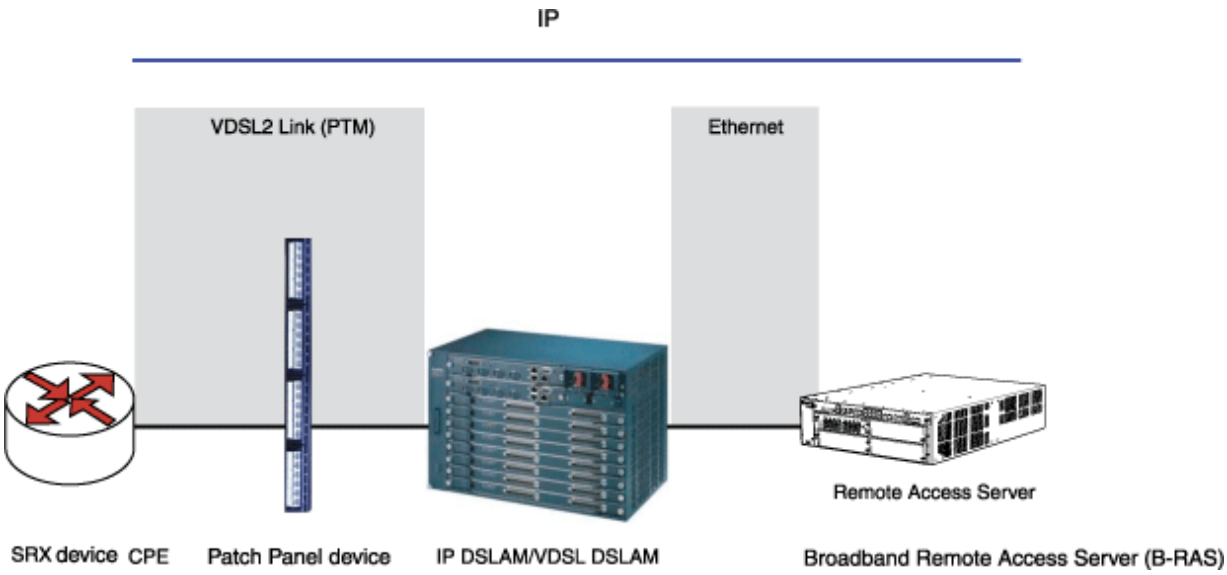
Feature	Description
Discrete multitone (DMT) modulation	• Separates a digital subscriber line signal to a usable frequency range of 256 frequency bands (or channels) with 4.3125 KHz each. • Uses the Fast Fourier Transform (FFT) algorithm for demodulation or modulation for increased speed.
Backward compatibility	• Backward compatible with most ADSL interface standards. • In ADSL fallback mode, VDSL2 operates on the ATM encapsulation interface in the first mile and uses the interface `at-1/0/0`. • Takes about 60 seconds to switch from VDSL2 to ADSL or from ADSL to VDSL2 operating modes.
Vectoring	• Employs coordination of line signals to reduce crosstalk levels to provide improved performance. • The ITU-T G.993.5 standard also known as G.vector, describes vectoring for VDSL2.
IPv6 Support	• Supports IPv6 on the DSL encapsulations like ATM physical interface encapsulations, atm-pvc, ethernet-over-atm, ethernet-over-atm, and ATM logical interface encapsulations except for atm-vc-mux and ppp-over-ether-over-atm-llc. • To configure IPv6 addresses on DSL interfaces in ATM or PTM mode, include the family protocol type as `inet6`.

For more information on supported features and profiles on VDSL2 interfaces, see1-Port VDSL2 Annex A Mini-Physical Interface Module (SRX-MP-1VDSL2-R).

VDSL2 Network Deployment Topology

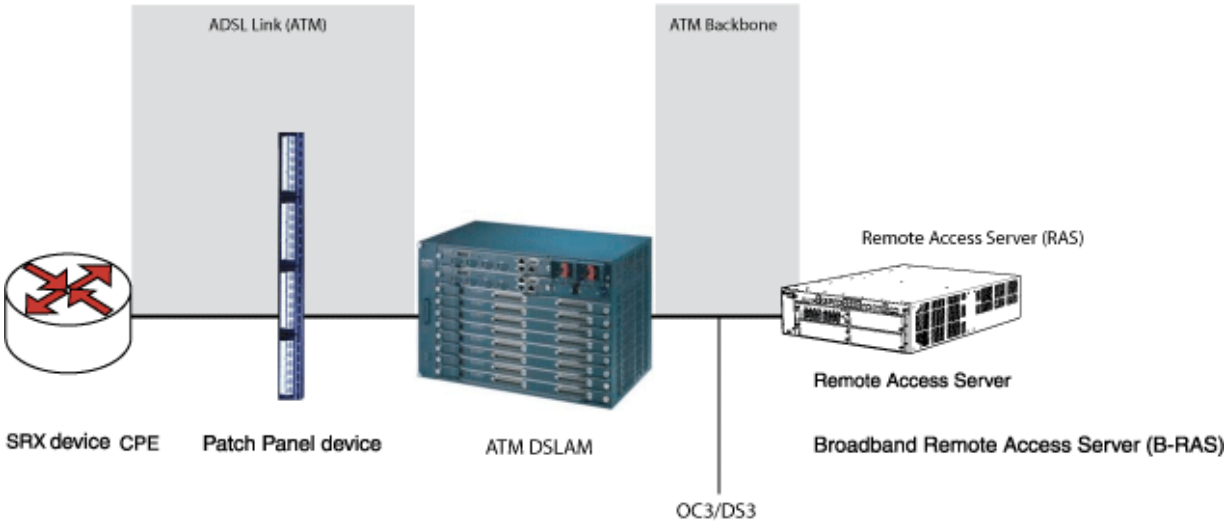
The VDSL2 interface uses either Gigabit Ethernet or fiber as second mile to connect to the Broadband Remote Access Server (B-RAS). Figure 9 on page 93 shows a typical VDSL2 network topology.

Figure 9: Typical VDSL2 End-to-End Connectivity and Topology Diagram



The ADSL interface uses either Gigabit Ethernet or OC3/DS3 ATM as the second mile to connect to the B-RAS. Figure 10 on page 93 shows a backward-compatible ADSL topology using ATM DSLAM.

Figure 10: Backward-Compatible ADSL Topology (ATM DSLAM)



Example: Configure VDSL2 Interface

IN THIS SECTION

In this example you configure the VDSL2 interface and VDSL2 interface on VDSL2 Mini-PIMs. On VDSL2 Mini-PIMs, the `pt-1/0/0` interface is created by default. You can switch to ADSL mode by configuring `at-1/0/0`. You can deactivate `pt-1/0/0` before you create `at-1/0/0` or deactivate `at-1/0/0` to create `pt-1/0/0`. Make sure that you have deleted the previous configurations on `pt-1/0/0` and `pp0`.

In this example:

1. Begin a new configuration on a VDSL2 Mini-PIM.
2. Deactivate previous interfaces and delete the old configuration.
3. Set the interfaces with the VDSL profile and the Layer 3 configuration for the end-to-end data path.
4. Configure the PPPoE on the `pt-1/0/0` interface with a static IP address or CHAP authentication with unnumbered IP address (PAP authentication or CHAP authentication).
5. Configure PPPoE on the `pt-1/0/0` interface with negotiated IP address (PAP authentication or CHAP authentication).

To configure VDSL2 Interfaces in ADSL mode:

1. Configure the ADSL interface for end-to-end data path.
2. Configure PPPoA on the `at-1/0/0` interface with a negotiated IP address and either PAP authentication or CHAP authentication.
3. Configure a static IP address and an unnumbered IP address (and either PAP authentication or CHAP authentication) for PPPoA on the `at-1/0/0` interface.
4. Configure PPPoE on the `at-1/0/0` interface with a negotiated IP address and either PAP authentication or CHAP authentication.

Table 19 on page 95 specifies the CLI quick configuration commands used for configuring VDSL2 interfaces.

Table 19: CLI Quick Configuration

Configuration Step	CLI Quick Configuration Commands
Configure the VDSL2 interface and enable VLAN tagging	**set interfaces pt-1/0/0 vdsl-options vdsl-profile auto** **set interfaces pt-1/0/0 vlan-tagging** **set interfaces pt-1/0/0 unit 0 vlan-id 100**
Begin a new configuration on a VDSL2 Mini-PIM	[edit] **deactivate interface pt-1/0/0** **deactivate interface at-1/0/0** **delete interface pt-1/0/0** **delete interface pp0**
Configure VDSL2 Mini-PIM for End-to-End Data Path	**set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a** **set interfaces pt-1/0/0 unit 0 family inet address 11.11.11.1/24**
Configure PPPoE on the pt-1/0/0 Interface with a Static IP Address	user@host# **set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a** user@host# **set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether** user@host# **set interfaces pp0 unit 0 ppp-options pap access-profile pap_prof local-name locky local-password india passive** user@host# **set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0 auto-reconnect 120 client** user@host# **set interfaces pp0 unit 0 family inet address 10.1.1.6/24** user@host# **set access profile pap_prof authentication-order password client cuttack pap-password india**

Table 19: CLI Quick Configuration *(Continued)*

Configuration Step	CLI Quick Configuration Commands
Configure PPPoE on the pt- Interface with a Static IP Address (CHAP Authentication)	user@host# **set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a** user@host# **set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether** user@host# **set interfaces pp0 unit 0 ppp-options chap default-chap-secret india local-name locky passive** user@host# **set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0 auto-reconnect 120 client** user@host# **set interfaces pp0 unit 0 family inet address 10.1.1.6/24**
Configure PPPoE on the pt-x/x/x Interface with Unnumbered IP (PAP Authentication)	user@host# **set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a** user@host# **set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether** user@host# **set interfaces lo0 unit 0 family inet address 10.1.1.24/32** user@host# **set interfaces pp0 unit 0 ppp-options pap access-profile pap_prof local-name locky local-password india passive** user@host# **set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0 auto-reconnect 120 client** user@host# **set interfaces pp0 unit 0 family inet unnumbered-address lo0.0 destination 10.1.1.1** user@host# **set access profile pap_prof authentication-order password client cuttack pap-password india**

Table 19: CLI Quick Configuration *(Continued)*

Configuration Step	CLI Quick Configuration Commands
Configure PPPoE on the pt-1/0/0 Interface with Unnumbered IP (CHAP Authentication)	user@host# **set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a** user@host# **set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether** user@host# **set interfaces lo0 unit 0 family inet address 10.1.1.24/32** user@host# **set interfaces pp0 unit 0 ppp-options chap default-chap-secret india local-name locky passive** user@host# **set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0 auto-reconnect 120 client** user@host# **set interfaces pp0 unit 0 family inet unnumbered-address lo0.0 destination 10.1.1.1**
Configure PPPoE on the pt-1/0/0 Interface with Negotiated IP (PAP Authentication)	user@host# **set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a** user@host# **set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether** user@host# **set interfaces pp0 unit 0 ppp-options pap access-profile my_prf local-name purple local-password <password> passive** user@host# **set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0 auto-reconnect 120 client** user@host# **set interfaces pp0 unit 0 family inet negotiate-address** user@host# **set access profile my_prf authentication-order password** user@host# **set access profile my_prf**

Table 19: CLI Quick Configuration *(Continued)*

Configuration Step	CLI Quick Configuration Commands
Configure PPPoE on the pt-1/0/0 Interface with Negotiated IP (CHAP Authentication)	user@host# **set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a** user@host# **set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether** user@host# **set interfaces pp0 unit 0 ppp-options chap default-chap-secret <password> local-name purple passive** user@host# **set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0 auto-reconnect 120 client** user@host# **set interfaces pp0 unit 0 family inet negotiate-address**

Configure the VDSL2 Interface and Enable VLAN Tagging

In this example, you create a VDSL2 interface called pt-1/0/0 and set the VDSL2 profile to auto. For more information on basic connectivity refer to Quick Start Guide and to configure network interfaces refer to "Example: Configure Ethernet Interface" on page 130. To configure the VDSL2 interfaces and enable VLAN tagging:

1. Create an interface.

```
[edit]
user@host# edit interfaces pt-1/0/0
```

2. Set the VDSL2 profile type.

```
[edit interfaces pt-1/0/0]
user@host# set vdsl-options vdsl-profile auto
```

3. Specify the logical unit to connect to the physical VDSL2 interface.

```
[edit interfaces pt-1/0/0]
user@host# set unit 0
```

4. Specify the family protocol type.

```
[edit interfaces pt-1/0/0]
user@host# set unit 0 family inet address 100.100.100.1/24
```

5. Enable VLAN tagging on the pt- interface.

```
[edit interfaces pt-1/0/0]
user@host# set interface pt-1/0/0 vlan-tagging
```

6. Specify the VLAN ID value.

```
[edit interfaces pt-1/0/0]
user@host# set interface pt-1/0/0 unit 0 vlan-id 100
```

7. Commit the configuration.

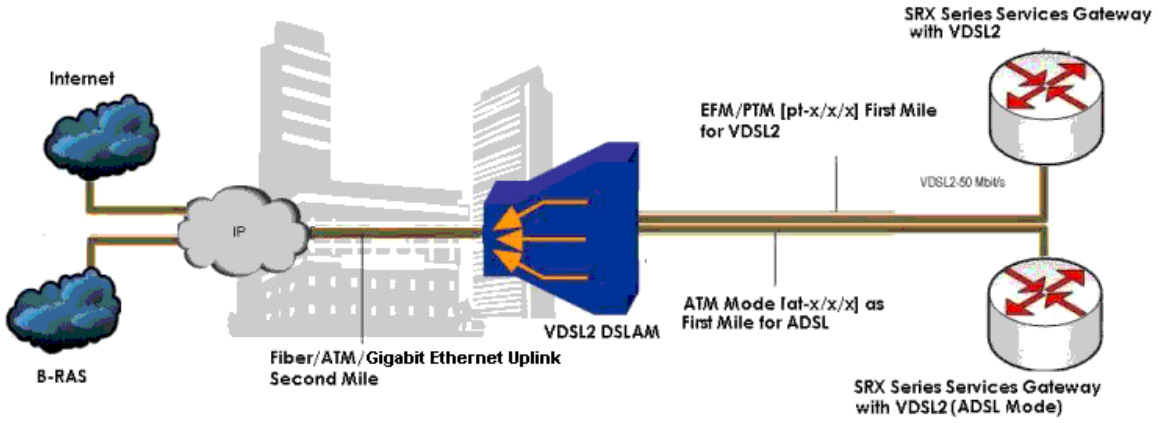
   VDSL2 is supported only on the pt- interface.The range of VLANs that can be configured is 0 to 4093.

Similarly, you can configure the VDSL2 interface on Annex B (integrated VDSL2 interfaces in ADSL backward compatible mode). After completing the configuration successfully, view the parameters by using the `show interfaces pt-1/0/0` command.

Configure VDSL2 Interface with VDSL2 Mini-PIMs

This example uses VDSL2 Mini-PIMs. Figure 11 on page 100 shows typical SRX Series Firewalls with VDSL2 Mini-PIM network connections.

Figure 11: SRX Series Firewall with VDSL2 Mini-PIMs in an End-to-End Deployment Scenario



To view the CLI quick configuration commands, see Table 19 on page 95.

To begin a new configuration on a VDSL2 Mini-PIM:

1. Deactivate any previous interfaces.

```
[edit]
user@host# deactivate interface pt-1/0/0
user@host# deactivate interface at-1/0/0
```

2. Delete any old configurations.

```
[edit]
user@host# delete interface pt-1/0/0
user@host# delete interface pp0
```

3. Commit the configuration.

Use the `show chassis fpc` command to see the output of the configuration.

Configure the VDSL2 Mini-PIM for End-to-End Data Path

To configure the VDSL2 Mini-PIM for end-to-end data path:

1. Configure the interfaces with the VDSL profile and the Layer 3 configuration for the end-to-end data path.

```
[edit]
user@host# set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a
user@host# set interfaces pt-1/0/0 unit 0 family inet address 11.11.11.1/24
```

2. Commit the configuration.

Use the `show interfaces pt-1/0/0` command to see the output of the configuration.

Configure PPPoE on the pt-1/0/0 Interface with a Static IP Address

To configure the PPPoE on the pt-1/0/0 interface with a static IP address:

1. Configure the VDSL options and encapsulation for the interface.

```
[edit]
user@host# set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a
user@host# set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether
```

2. Configure the PPP options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 ppp-options pap access-profile pap_prof
user@host# set interfaces pp0 unit 0 ppp-options pap local-name locky
user@host# set interfaces pp0 unit 0 ppp-options pap local-password india
user@host# set interfaces pp0 unit 0 ppp-options pap passive
```

3. Configure the PPPoE options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0
user@host# set interfaces pp0 unit 0 pppoe-options auto-reconnect 120
user@host# set interfaces pp0 unit 0 pppoe-options client
```

4. Configure the IP address for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 family inet address 10.1.1.6/24
```

5. Configure the access profile for the interface.

```
[edit]
user@host# set access profile pap_prof authentication-order password
user@host# set access profile pap_prof client cuttack pap-password india
```

6. Commit the configuration.

Use the `show interfaces pp0`, `show interfaces pt-1/0/0` and `show access profile pap_prof` commands to see the output of the configuration.

Configure PPPoE on the pt-1/0/0 Interface with a Static IP Address (CHAP Authentication)

To configure the PPPoE on the pt-1/0/0 interface with a static IP address (CHAP authentication):

1. Configure the VDSL options and encapsulation for the interface.

```
[edit]
user@host# set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a
user@host# set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether
```

2. Configure the PPP options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 ppp-options chap default-chap-secret india
user@host# set interfaces pp0 unit 0 ppp-options chap local-name locky
user@host# set interfaces pp0 unit 0 ppp-options chap passive
```

3. Configure the PPPoE options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0
user@host# set interfaces pp0 unit 0 pppoe-options auto-reconnect 120
user@host# set interfaces pp0 unit 0 pppoe-options client
```

4. Configure the IP address for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 family inet address 10.1.1.6/24
```

5. Commit the configuration.

Use the `show interfaces pt-1/0/0` and `show interfaces pp0` commands to see the output of the configuration.

Configure PPPoE on the pt-x/x/x Interface with Unnumbered IP (PAP Authentication)

To configure PPPoE on the pt-1/0/0 interface with unnumbered IP (PAP authentication):

1. Configure the VDSL options and encapsulation for the interface.

```
[edit]
user@host# set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a
user@host# set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether
```

2. Configure the IP address for the interface.

```
[edit]
user@host# set interfaces lo0 unit 0 family inet address 10.1.1.24/32
```

3. Configure the PPP options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 ppp-options pap access-profile pap_prof
user@host# set interfaces pp0 unit 0 ppp-options pap local-name locky
user@host# set interfaces pp0 unit 0 ppp-options pap local-password india
user@host# set interfaces pp0 unit 0 ppp-options pap passive
```

4. Configure the PPPoE options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0
user@host# set interfaces pp0 unit 0 pppoe-options auto-reconnect 120
user@host# set interfaces pp0 unit 0 pppoe-options client
```

5. Configure the unnumbered address and destination for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 family inet unnumbered-address lo0.0
user@host# set interfaces pp0 unit 0 family inet unnumbered-address destination 10.1.1.1
```

6. Configure the access profile for the interface.

```
[edit]
user@host# set access profile pap_prof authentication-order password
user@host# set access profile pap_prof client cuttack pap-password india
```

7. Commit the configuration.

Use the `show interfaces lo0`, `show interfaces pt-1/0/0`, and `show interfaces pp0` commands to see the output of the configuration.

Configuring PPPoE on the pt-1/0/0 Interface with Unnumbered IP (CHAP Authentication)

To configure PPPoE on the pt-1/0/0 interface with unnumbered IP (CHAP authentication):

1. Configure the VDSL options and encapsulation for the interface.

```
[edit]
user@host# set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a
user@host# set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether
```

2. Configure the IP address for the interface.

```
[edit]
user@host# set interfaces lo0 unit 0 family inet address 10.1.1.24/32
```

3. Configure the PPP options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 ppp-options chap default-chap-secret india
user@host# set interfaces pp0 unit 0 ppp-options chap local-name locky
user@host# set interfaces pp0 unit 0 ppp-options chap passive
```

4. Configure the PPPoE options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0
user@host# set interfaces pp0 unit 0 pppoe-options auto-reconnect 120
user@host# set interfaces pp0 unit 0 pppoe-options client
```

5. Configure the unnumbered address and destination for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 family inet unnumbered-address lo0.0
user@host# set interfaces pp0 unit 0 family inet unnumbered-address destination 10.1.1.1
```

6. Commit the configuration.

Use the `show interfaces pp0`, `show interfaces pt-1/0/0`, and `show interfaces lo0` commands to see the output of the configuration.

Configure PPPoE on the pt-1/0/0 Interface with Negotiated IP (PAP Authentication)

To configure PPPoE on the pt-1/0/0 interface with negotiated IP (PAP authentication):

1. Configure the VDSL options and encapsulation for the interface.

```
[edit]
user@host# set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a
user@host# set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether
```

2. Configure the PPP options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 ppp-options pap access-profile my_prf
user@host# set interfaces pp0 unit 0 ppp-options pap local-name purple
user@host# set interfaces pp0 unit 0 ppp-options pap local-password <password>
user@host# set interfaces pp0 unit 0 ppp-options pap passive
```

3. Configure the PPPoE options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0
```

```
user@host# set interfaces pp0 unit 0 pppoe-options auto-reconnect 120
user@host# set interfaces pp0 unit 0 pppoe-options client
```

4. Configure the negotiated IP address for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 family inet negotiate-address
```

5. Configure the access profile for the interface.

```
[edit]
user@host# set access profile my_prf authentication-order password
user@host# set access profile my_prf
```

6. Commit the configuration.

Use the `show interfaces pt-1/0/0`, `show interfaces pp0`, and `show access profile my_prf` commands to see the output of the configuration.

Configure PPPoE on the pt-1/0/0 Interface with Negotiated IP (CHAP Authentication)

To configure PPPoE on the pt-1/0/0 interface with negotiated IP (CHAP authentication):

1. Configure the VDSL options and encapsulation for the interface.

```
[edit]
user@host# set interfaces pt-1/0/0 vdsl-options vdsl-profile 17a
user@host# set interfaces pt-1/0/0 unit 0 encapsulation ppp-over-ether
```

2. Configure the PPP options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 ppp-options chap default-chap-secret <password>
user@host# set interfaces pp0 unit 0 ppp-options chap local-name purple
user@host# set interfaces pp0 unit 0 ppp-options chap passive
```

3. Configure the PPPoE options for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0.0
user@host# set interfaces pp0 unit 0 pppoe-options auto-reconnect 120
user@host# set interfaces pp0 unit 0 pppoe-options client
```

4. Configure the negotiated IP address for the interface.

```
[edit]
user@host# set interfaces pp0 unit 0 family inet negotiate-address
```

5. Commit the configuration.

Use the `show interfaces pp0` and `show interfaces pt-1/0/0` commands to see the output of the configuration. Similarly, you can configure the integrated VDSL2 interfaces, Annex B, in ADSL backward compatible mode by using the `show interfaces pt-1/0/0` command.

Verification

IN THIS SECTION

Purpose

Display information about the parameters configured on the VDSL2 interface.

Action

- To display information about the parameters configured on VDSL2 Interface connected to the DSLAM, operating in Annex A and display details of VLAN tagging:

```
user@host>  show interfaces pt-1/0/0
```

```
Physical interface: pt-1/0/0, Enabled, Physical link is Up
 Interface index: 146, SNMP ifIndex: 524, Generation: 149
 Type: PTM, Link-level type: Ethernet, MTU: 1496, VDSL mode, Speed:    45440kbps
 Speed: VDSL2
 Device flags : Present Running
 Link flags : None
 CoS queues : 8 supported, 8 maximum usable queues
 Hold-times : Up 0 ms, Down 0 ms
 Current address: 00:b1:7e:85:84:ff
 Last flapped : 2009-10-18 11:56:50 PDT (12:32:49 ago)
 Statistics last cleared: 2009-10-19 00:29:37 PDT (00:00:02 ago)
 Traffic statistics:
  Input bytes : 22438962 97070256 bps
  Output bytes : 10866024 43334088 bps
  Input packets: 15141 8187 pps
  Output packets: 7332 3655 pps
Input errors:
   Errors: 0, Drops: 0, Policed discards: 0, L3 incompletes: 0,
   L2 channel errors: 0, L2 mismatch timeouts: 0, Resource errors: 0
 Output errors:
  Carrier transitions: 0, Errors: 0, Drops: 0, Aged packets: 0, MTU  errors: 0,
  Resource errors: 0
 Egress queues: 8 supported, 4 in use
 Queue counters: Queued packets Transmitted packets Dropped packets
    0 best-effort 6759 6760 0
    1 expedited-fo 0 0 0
    2 assured-forw 0 0 0
    3 network-cont 0 0 0
VDSL alarms : None
VDSL defects : None
VDSL media: Seconds Count State
   LOF 0 0 OK
   LOS 0 0 OK
   LOM 0 0 OK
```

```
   LOP 0 0 OK
   LOCDI 0 0 OK
   LOCDNI 0 0 OK
VDSL status:
  Modem status : Showtime (Profile-17a)
  VDSL profile : Profile-17a Annex A
  Last fail code: None
  Subfunction : 0x00
  Seconds in showtime : 45171
continue.....................................................
.....................................................
```

Similarly, you can verify the VDSL2 interface on Annex B mode by using the `show interfaces pt-1/0/0` command.

```
user@host>  show interfaces pt-1/0/0
```

```
vlan-tagging;
vdsl-options {
vdsl-profile auto;
}
unit 0 {
vlan-id 100;
Family inet {
address 100.100.100.1/24;
}
```

- Verify the FPC status by entering the `show chassis fpc` command. The VDSL2 Mini-PIM is installed in the first slot of the SRX320 device chassis; therefore, use `fpc 1`. For SRX340 devices, use the FPCs `fpc 1`, `fpc 2`, `fpc 3`, or `fpc 4`.

```
user@host>  show chassis fpc
```

```
Temp CPU Utilization (%) Memory Utilization
(%)
Slot State (C) Total Interrupt DRAM (MB) Heap Buffer
```

```
0 Online -------------------- CPU less FPC --------------------
1 Online -------------------- CPU less FPC --------------------
```

- Verify the status of interface, modem status, time in seconds and VDSL profile of DSLAM by using the `run show interface pt-1/0/0`.

```
user@host> show interface pt-1/0/0
```

```
Physical interface: pt-1/0/0, Enabled, Physical link is Up
 Interface index: 146, SNMP ifIndex: 524, Generation: 149
 Type: PTM, Link-level type: Ethernet, MTU: 1496, VDSL mode, Speed:    45440kbps
 Speed: VDSL2
 Device flags : Present Running
 Link flags : None
 CoS queues : 8 supported, 8 maximum usable queues
 Hold-times : Up 0 ms, Down 0 ms
 Current address: 00:b1:7e:85:84:ff
 Last flapped : 2009-10-18 11:56:50 PDT (12:32:49 ago)
 Statistics last cleared: 2009-10-19 00:29:37 PDT (00:00:02 ago)
 Traffic statistics:
  Input bytes : 22438962 97070256 bps
  Output bytes : 10866024 43334088 bps
  Input packets: 15141 8187 pps
  Output packets: 7332 3655 pps
Input errors:
   Errors: 0, Drops: 0, Policed discards: 0, L3 incompletes: 0,
   L2 channel errors: 0, L2 mismatch timeouts: 0, Resource errors: 0
 Output errors:
  Carrier transitions: 0, Errors: 0, Drops: 0, Aged packets: 0, MTU  errors: 0,
  Resource errors: 0
 Egress queues: 8 supported, 4 in use
 Queue counters: Queued packets Transmitted packets Dropped packets
    0 best-effort 6759 6760 0
    1 expedited-fo 0 0 0
    2 assured-forw 0 0 0
    3 network-cont 0 0 0
VDSL alarms : None
VDSL defects : None
VDSL media: Seconds Count State
   LOF 0 0 OK
```

```
    LOS 0 0 OK
    LOM 0 0 OK
    LOP 0 0 OK
    LOCDI 0 0 OK
    LOCDNI 0 0 OK
VDSL status:
  Modem status : Showtime (Profile-17a)
  VDSL profile : Profile-17a Annex A
  Last fail code: None
  Subfunction : 0x00
  Seconds in showtime : 45171
continue................................................
.................................................
```

- To display all the parameters configured on VDSL2 Mini-PIM for End-to-End Data Path

```
user@host> show interfaces pt-1/0/0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pt-1/0/0                up    up
pt-1/0/0.0              up    up   inet     11.11.11.1/24

[edit]
user@host# run ping 11.11.11.2 count 1000 rapid
PING 11.11.11.2 (11.11.11.2): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!
- 11.11.11.2 ping statistics ---
1000 packets transmitted, 1000 packets received, 0% packet loss
round-trip min/avg/max/stddev = 16.109/17.711/28.591/2.026 ms
```

```
user@host> show interfaces pt-1/0/0 extensive
```

```
Physical interface: pt-1/0/0, Enabled, Physical link is Up
  Interface index: 146, SNMP ifIndex: 524, Generation: 197
  Type: PTM, Link-level type: Ethernet, MTU: 1496, VDSL mode, Speed: 45440kbps
  Speed: VDSL2
  Device flags   : Present Running
```

```
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:b1:7e:85:84:ff
  Last flapped   : 2009-10-28 00:36:29 PDT (00:12:03 ago)
  Statistics last cleared: 2009-10-28 00:47:56 PDT (00:00:36 ago)
  Traffic statistics:
   Input  bytes  :                84000                    0 bps
   Output bytes  :               138000                    0 bps
   Input  packets:                 1000                    0 pps
   Output packets:                 1000                    0 pps
  Input errors:
    Errors: 0, Drops: 0, Policed discards: 0, L3 incompletes: 0, L2 channel errors: 0, L2
mismatch timeouts: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, Aged packets: 0, MTU errors: 0, Resource
errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort                1000                 1000                    0
    1 expedited-fo                  0                    0                    0
    2 assured-forw                  0                    0                    0
    3 network-cont                  0                    0                    0
  VDSL  alarms   : None
  VDSL  defects  : None
  VDSL media:            Seconds        Count  State
    LOF                         0            0  OK
    LOS                         0            0  OK
    LOM                         0            0  OK
    LOP                         0            0  OK
    LOCDI                       0            0  OK
    LOCDNI                      0            0  OK
  VDSL status:
    Modem status  : Showtime (Profile-17a)
    VDSL profile  : Profile-17a Annex A
    Last fail code: None
```

- To display the PPPoE on the pt-1/0/0 Interface with a Static IP Address.

```
user@host> show interfaces pp0
```

```
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 510
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
    Input packets : 0
    Output packets: 0

  Logical interface pp0.0 (Index 71) (SNMP ifIndex 522)
    Flags: Hardware-Down Point-To-Point SNMP-Traps 0x0 Encapsulation: PPPoE
    PPPoE:
      State: SessionDown, Session ID: None,
      Configured AC name: None, Service name: None,
      Auto-reconnect timeout: 120 seconds, Idle timeout: Never,
      Underlying interface: pt-1/0/0.0 (Index 69)
    Input packets : 57
    Output packets: 56
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 22 (00:00:40 ago), Output: 25 (00:00:04 ago)
  LCP state: Down
```

```
user@host> show interfaces pt-1/0/0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pt-1/0/0                up    up
pt-1/0/0.0              up    up

[edit]
user@host# run show interfaces pp0 terse
Interface               Admin Link Proto    Local                 Remote
pp0                     up    up
pp0.0                   up    up   inet     10.1.1.6/24
```

```
[edit]
user@host# run ping 10.1.1.1 count 100 rapid
PING 10.1.1.1 (10.1.1.1): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!
- 10.1.1.1 ping statistics ---
100 packets transmitted, 100 packets received, 0% packet loss
round-trip min/avg/max/stddev = 14.669/15.649/21.655/1.740 ms
```

- To display PPPoE on the pt-1/0/0 Interface with a Static IP Address (CHAP Authentication)

```
user@host> show interfaces pp0
```

```
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 510
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
    Input packets : 0
    Output packets: 0

  Logical interface pp0.0 (Index 70) (SNMP ifIndex 522)
    Flags: Point-To-Point SNMP-Traps 0x0 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 31,
      Session AC name: cuttack, Remote MAC address: 00:03:6c:c8:8c:55,
      Configured AC name: None, Service name: None,
      Auto-reconnect timeout: 120 seconds, Idle timeout: Never,
      Underlying interface: pt-1/0/0.0 (Index 69)
    Input packets : 12
    Output packets: 10
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 1 (00:00:08 ago), Output: 0 (never)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls: Not-configured
  CHAP state: Success
  PAP state: Closed
```

```
    Security: Zone: Null
    Protocol inet, MTU: 1492
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.1.1/24, Local: 10.1.1.6
```

```
user@host> show interfaces pt-1/0/0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pt-1/0/0                up    up
pt-1/0/0.0              up    up
```

```
user@host> show interfaces pp0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pp0                     up    up
pp0.0                   up    up   inet     10.1.1.6/24
```

```
user@host> ping 10.1.1.1 count 100 rapid
```

```
PING 10.1.1.1 (10.1.1.1): 56 data bytes

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!
--- 10.1.1.1 ping statistics ---
100 packets transmitted, 100 packets received, 0% packet loss
round-trip min/avg/max/stddev = 14.608/15.466/25.939/1.779 ms
```

- To display the PPPoE on the pt-1/0/0 Interface with Unnumbered IP (PAP Authentication)

```
user@host> show interfaces pp0
```

```
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 510
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
    Input packets : 0
    Output packets: 0

  Logical interface pp0.0 (Index 72) (SNMP ifIndex 522)
    Flags: Point-To-Point SNMP-Traps 0x0 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 33,
      Session AC name: cuttack, Remote MAC address: 00:03:6c:c8:8c:55,
      Configured AC name: None, Service name: None,
      Auto-reconnect timeout: 120 seconds, Idle timeout: Never,
      Underlying interface: pt-1/0/0.0 (Index 69)
    Input packets : 22
    Output packets: 20
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 1 (00:00:08 ago), Output: 0 (never)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls: Not-configured
  CHAP state: Closed
  PAP state: Success
    Security: Zone: Null
    Protocol inet, MTU: 1492
      Flags: None
```

```
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.1.1.1, Local: 10.1.1.24
```

```
user@host> show interfaces pt-1/0/0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pt-1/0/0                up    up
pt-1/0/0.0              up    up
```

```
user@host> show interfaces pp0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pp0                     up    up
pp0.0                   up    up   inet     10.1.1.24           --> 10.1.1.1
```

```
user@host> ping 10.1.1.1 count 100 rapid
```

```
PING 10.1.1.1 (10.1.1.1): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!
--- 10.1.1.1 ping statistics ---
100 packets transmitted, 100 packets received, 0% packet loss
round-trip min/avg/max/stddev = 14.584/15.503/21.204/1.528 ms
```

- To display the PPPoE on the pt-1/0/0 Interface with Unnumbered IP (CHAP Authentication)

```
user@host> show interfaces pp0
```

```
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 510
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532
  Device flags   : Present Running
```

```
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
    Input packets : 0
    Output packets: 0

  Logical interface pp0.0 (Index 70) (SNMP ifIndex 522)
    Flags: Point-To-Point SNMP-Traps 0x0 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 35,
      Session AC name: cuttack, Remote MAC address: 00:03:6c:c8:8c:55,
      Configured AC name: None, Service name: None,
      Auto-reconnect timeout: 120 seconds, Idle timeout: Never,
      Underlying interface: pt-1/0/0.0 (Index 69)
    Input packets : 25
    Output packets: 22
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 2 (00:00:10 ago), Output: 2 (00:00:02 ago)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls: Not-configured
  CHAP state: Success
  PAP state: Closed
    Security: Zone: Null
    Protocol inet, MTU: 1492
      Flags: None
```

```
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.1.1.1, Local: 10.1.1.24
```

```
user@host> show interfaces pt-1/0/0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pt-1/0/0                up    up
pt-1/0/0.0              up    up
```

```
user@host> show interfaces pp0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pp0                     up    up
pp0.0                   up    up   inet     10.1.1.24           --> 10.1.1.1
```

```
user@host> ping 10.1.1.1 count 100 rapid
```

```
PING 10.1.1.1 (10.1.1.1): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!
-- 10.1.1.1 ping statistics ---
100 packets transmitted, 100 packets received, 0% packet loss
round-trip min/avg/max/stddev = 14.585/16.025/22.354/2.019 ms
```

- To display PPPoE on the pt-1/0/0 Interface with Negotiated IP (PAP Authentication)

```
user@host> show interfaces pp0
```

```
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 510
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532
  Device flags   : Present Running
```

```
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
    Input packets : 0
    Output packets: 0

  Logical interface pp0.0 (Index 72) (SNMP ifIndex 522)
    Flags: Point-To-Point SNMP-Traps 0x0 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 4,
      Session AC name: belur, Remote MAC address: 00:90:1a:43:18:d1,
      Configured AC name: None, Service name: None,
      Auto-reconnect timeout: 120 seconds, Idle timeout: Never,
      Underlying interface: pt-1/0/0.0 (Index 69)
    Input packets : 18
    Output packets: 18
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 0 (never), Output: 11 (00:00:01 ago)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls: Not-configured
  CHAP state: Closed
  PAP state: Success
    Security: Zone: Null
    Protocol inet, MTU: 1474
      Flags: Negotiate-Address
```

```
      Addresses, Flags: Kernel Is-Preferred Is-Primary
        Destination: 12.12.12.1, Local: 12.12.12.11
```

```
user@host> show interfaces pt-1/0/0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pt-1/0/0                up    up
pt-1/0/0.0              up    up
```

```
user@host> show interfaces pp0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pp0                     up    up
pp0.0                   up    up   inet     12.12.12.11         --> 12.12.12.1
```

```
user@host> ping 12.12.12.1 count 100 rapid
```

```
PING 12.12.12.1 (12.12.12.1): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!
--- 12.12.12.1 ping statistics ---
100 packets transmitted, 100 packets received, 0% packet loss
round-trip min/avg/max/stddev = 16.223/17.692/24.359/2.292 ms
```

- To display the PPPoE on the pt-1/0/0 Interface with Negotiated IP (CHAP Authentication)

```
user@host> show interfaces pp0
```

```
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 510
  Type: PPPoE, Link-level type: PPPoE, MTU: 1532
  Device flags   : Present Running
```

```
Interface flags: Point-To-Point SNMP-Traps
Link type      : Full-Duplex
Link flags     : None
  Input packets : 0
  Output packets: 0

Logical interface pp0.0 (Index 70) (SNMP ifIndex 522)
  Flags: Point-To-Point SNMP-Traps 0x0 Encapsulation: PPPoE
  PPPoE:
    State: SessionUp, Session ID: 8,
    Session AC name: belur, Remote MAC address: 00:90:1a:43:18:d1,
    Configured AC name: None, Service name: None,
    Auto-reconnect timeout: 120 seconds, Idle timeout: Never,
    Underlying interface: pt-1/0/0.0 (Index 69)
  Input packets : 12
  Output packets: 11
Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
Keepalive: Input: 0 (never), Output: 4 (00:00:03 ago)
LCP state: Opened
NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls: Not-configured
CHAP state: Success
PAP state: Closed
  Security: Zone: Null
  Protocol inet, MTU: 1474
    Flags: Negotiate-Address
```

```
      Addresses, Flags: Kernel Is-Preferred Is-Primary
        Destination: 12.12.12.1, Local: 12.12.12.12
```

```
user@host> show interfaces pt-1/0/0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pt-1/0/0                up    up
pt-1/0/0.0              up    up
```

```
user@host> show interfaces pp0 terse
```

```
Interface               Admin Link Proto    Local                 Remote
pp0                     up    up
pp0.0                   up    up   inet     12.12.12.12         --> 12.12.12.1
```

```
user@host> ping 12.12.12.1 count 100 rapid
```

```
PING 12.12.12.1 (12.12.12.1): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!
--- 12.12.12.1 ping statistics ---
100 packets transmitted, 100 packets received, 0% packet loss
round-trip min/avg/max/stddev = 16.168/17.452/23.299/2.016 ms
```

RELATED DOCUMENTATION

1-Port VDSL2 (Annex A) Mini-Physical Interface Module Supported Profiles

3

CHAPTER

Configuring Ethernet Interfaces

IN THIS CHAPTER

Ethernet Interfaces

IN THIS SECTION

Learn about Ethernet technology used to broadcast traffic on security devices, static ARP entries, creating and deleting the Ethernet interface, and enabling and disabling the promiscuous mode on these interfaces. Also learn about Aggregated Ethernet Interfaces

Ethernet Interfaces Overview

IN THIS SECTION

Ethernet is a Layer 2, point-to multipoint technology that operates in a shared bus topology, supports broadcast transmission, and has distributed access control.

In a shared bus topology, all devices connect to a single, shared physical link through which all data transmissions are sent. The devices within a single Ethernet topology make up a broadcast domain.

The physical hardware does not provide information to the sender about incoming and lost traffic. Higher layer protocols such as TCP/IP can provide this type of notification.

Table 20: Types of Ethernet Interfaces

Types	Description
Ethernet Access Control and Transmission	• Ethernet's access control is distributed. • Uses carrier-sense multiple access with collision detection (CSMA/CD) mechanism. • If there is no transmission host begins transmitting its own data. • Length of each transmission is determined by fixed Ethernet packet size. • Enforces a minimum idle time between transmissions. • Ensures there is no interruption in sending and receiving traffic.

Collisions and Detection	• Delay, or latency, in transmitting traffic results in collision of two electrical signals.
	• Signals are scrambled so that both transmissions are effectively lost .
	• Two types include: Collision detection and Backoff Algorithm
	  • Collision detection refers to link monitoring while the devices are transmitting data. The device transmits data during the idle state on the wire.
	  • Binary exponential backoff algorithm helps each device, sending a colliding transmission randomly, select a value within a range. The value represents the number of transmission times that the device must wait before retransmitting its data. Each time a collision occurs, the range of values doubles.

Table 20: Types of Ethernet Interfaces *(Continued)*

Types	Description

Collision Domains and LAN Segments	• Multiple collision domains can be interconnected by repeaters, bridges, and switches if the length of an Ethernet cable restrict the length of a LAN segment.
	• Repeaters are electronic devices that act on analog signals and relay all electronic signals. Ethernet specification restricts the number of repeaters to two. A single repeater can double the distance between two devices on an Ethernet network.
	• Bridges and switches combine LAN segments into a single Ethernet network by using multiple ports to connect the physical wires in each segment.
	• Bridges provide more management and interface ports.
	• Bridge tracks the source MAC address of the packets and stores the addresses and their associated input ports in an interface table.
	• The bridge examines its interface table and takes one of the following actions:
	  • If the destination address does not match an interface table address, the bridge transmits the packet to all hosts on the network using the Ethernet broadcast address.
	  • If the destination address matches the port with receiving packet, the bridge or switch discards the packet. The bridge does not need to retransmit it.
	  • If the destination address maps to a port other than the one through which the packet was received, the bridge transmits the packet through the appropriate port to the corresponding LAN segment.

	• Combination of all the LAN segments within an Ethernet network is called broadcast domain. • When you use a bridge or switch, the broadcast domain consists of the entire LAN.

Table 21 on page 128

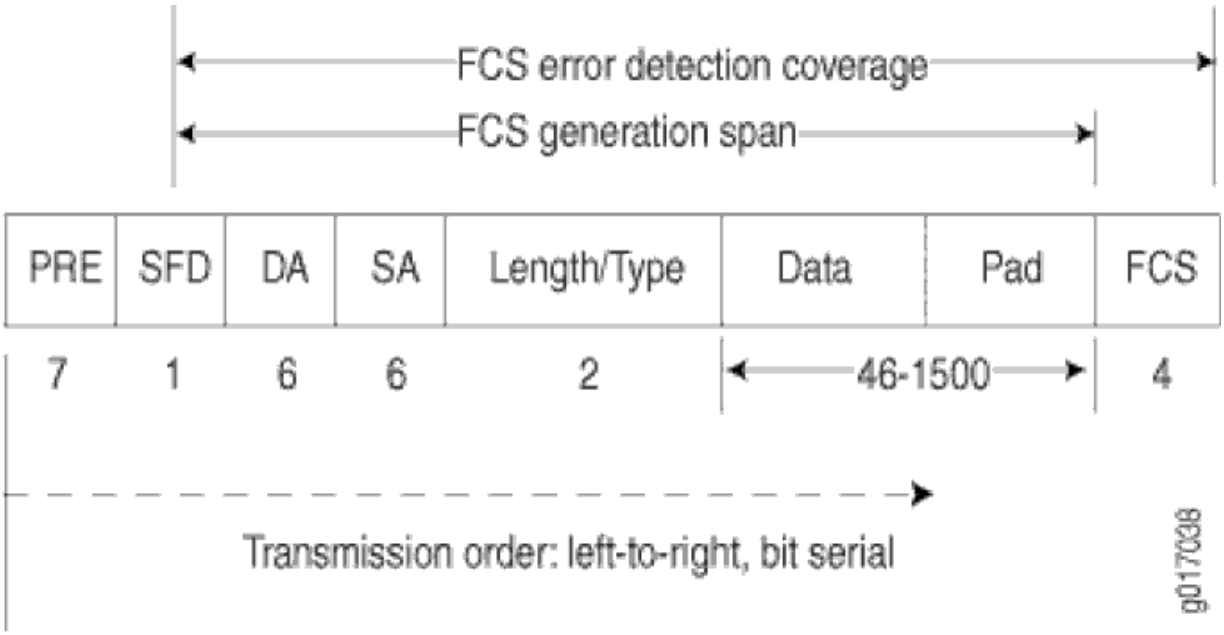
Table 21: Collision Backoff Algorithm Rounds

Round	Size of Set	Elements in the Set
1	2	{0,1}
2	4	{0,1,2,3}
3	8	{0,1,2,3,...,7}
4	16	{0,1,2,3,4,...,15}
5	32	{0,1,2,3,4,5,...,31}
6	64	{0,1,2,3,4,5,6,...,63}
7	128	{0,1,2,3,4,5,6,7,...,127}
8	256	{0,1,2,3,4,5,6,7,8,...,255}
9	512	{0,1,2,3,4,5,6,7,8,9,...,511}
10	1024	{0,1,2,3,4,5,6,7,8,9,10,...,1023}

Ethernet Frames

Data is transmitted through an Ethernet network in frames. The frames are of variable length, ranging from 64 octets to 1518 octets, including the header, payload, and cyclic redundancy check (CRC) value.Figure 12 on page 129 shows the Ethernet frame format.

Figure 12: Ethernet Frame Format



Ethernet frames have the following fields:

- The preamble (PRE) field is 7 octets of alternating 0s and 1s. The predictable format in the preamble allows receiving interfaces to synchronize themselves to the data being sent. The preamble is followed by a 1-octet start-of-frame delimiter (SFD).
- The destination address (DA) and source address (SA) fields contain the 6-octet (48-bit) MAC addresses for the destination and source ports on the network. These Layer 2 addresses uniquely identify the devices on the LAN.
- The Length/Type field is a 2-octet field that either indicates the length of the frame's data field or identifies the protocol stack associated with the frame. Here are some common frame types:
  - AppleTalk—`0x809B`
  - AppleTalk ARP—`0x80F3`
  - DECnet—`0x6003`
  - IP—`0x0800`
  - IPX—`0x8137`
  - Loopback—`0x9000`
  - XNS—`0x0600`
- The Data field contains the packet payload.

- The frame check sequence (FCS) is a 4-octet field that contains the calculated CRC value. This value is calculated by the originating host and appended to the frame. When it receives the frames, the receiving host calculates the CRC and checks it against this appended value to verify the integrity of the received frame.
- On SRX650 devices, MAC pause frame and FCS error frame counters are not supported for the interfaces ge-0/0/0 through ge-0/0/3. (Platform support depends on the Junos OS Release in your installation.)

Promiscuous Mode

- When you enable promiscuous mode on a Layer 3 Ethernet interface, all received packets on are sent to the central point or Services Processing Unit (SPU) regardless of the destination MAC address of the packet.
- You can also enable promiscuous mode on chassis cluster redundant Ethernet interfaces and aggregated Ethernet interfaces.
- If you enable promiscuous mode on a redundant Ethernet interface, it is enabled on any child physical interfaces. If you enable promiscuous mode on an aggregated Ethernet interface, it is enabled on all member interfaces.
- Promiscuous mode function is supported on 1-Gigabit, 10-Gigabit, 40-Gigabit, and 100-Gigabit Ethernet interfaces on the I/O cards (IOCs) and the SRX5000 line Module Port Concentrator (SRX5K-MPC).
- By default, an interface enables MAC filtering. You can configure promiscuous mode on the interface to disable MAC filtering. When you delete the configuration, the interface will perform MAC filtering again.
- You can change the interface MAC address when the interface is operating in promiscuous mode. When the interface is operating in normal mode, the MAC filtering function on the IOC uses the new MAC address to filter the packets.

Example: Configure Ethernet Interface

IN THIS SECTION

Overview

Table describes the steps to create and (optional) delete Ethernet interfaces on your routing device.

Table 22: Ethernet Interfaces Configuration

Configuration Step	Command
Step 1: Create the Ethernet interface and set the logical interface.	[edit] user@host# **edit interfaces ge-1/0/0 unit 0**
Step 2: If you are done configuring the device, commit the configuration.	[edit] user@host# **commit**
Step 3: (Optional) Specify the interface you want to delete.	[edit] user@host# **delete interfaces ge-1/0/0**
Step 4: If you are done configuring the device, commit the configuration.	[edit] user@host# **commit**

Example: Configuring Promiscuous Mode on the SRX5K-MPC

IN THIS SECTION

This example shows how to configure promiscuous mode on an SRX5K-MPC interface in an SRX5600 to disable MAC address filtering.

CLI Quick Configuration

Below table specifies the CLI quick configuration commands used for configuring and disabling promiscuous mode on SRX5K-MPC interface .

Table 23: CLI Quick Configuration

Configuration Step	CLI Quick Configuration Commands
Configure promiscuous mode on the interface	**set interfaces et-4/0/0 unit 0 family inet address 10.1.1.1/24** **set interfaces et-4/0/0 promiscuous-mode**
Disable promiscuous mode on an interface	user@host# **delete interfaces et-4/0/0 promiscuous-mode**

Configure Promiscuous Mode on an Interface

Below table describes the step-by-step to configure promiscuous mode on an interface on your security device.

Table 24: Promiscuous Mode Configuration

Configuration Step	Command
Step 1: Configure the ingress interface.	[edit interfaces] user@host# **set et-4/0/0 unit 0 family inet address 10.1.1.1/24**
Step 2: Enable promiscuous mode on the interface.	[edit interfaces] user@host# **set et-4/0/0 promiscuous-mode**

Table 24: Promiscuous Mode Configuration *(Continued)*

Configuration Step	Command
Step 3: (Optional) Disable promiscuous mode on the interface.	`[edit]` `user@host# delete interfaces et-4/0/0 promiscuous-mode`

Use the `show interfaces` command to see the output of the configuration.

Verification

Purpose

Verify that promiscuous mode is enabled, its status, on the interface and disabled on the interface.

Action

- To display information about the parameters configured on promiscuous mode Interface.

```
user@host> show interfaces
```

```
Physical interface: et-4/0/0, Enabled, Physical link is Up
  Interface index: 137, SNMP ifIndex: 511
  Link-level type: Ethernet, MTU: 1518, Speed: 100Gbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled
  Device flags   : Present Running
  Interface flags: Promiscuous SNMP-Traps Internal: 0x4000
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 2c:21:72:3a:05:28, Hardware address: 2c:21:72:3a:05:28
  Last flapped   : 2014-01-17 14:44:53 PST (5d 06:30 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : None
  Active defects : None
  PCS statistics                      Seconds
    Bit errors                             0
    Errored blocks                         0
```

```
  Logical interface et-4/0/0.0 (Index 71) (SNMP ifIndex 513)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1351 ]  Encapsulation: ENET2
    Input packets : 0
    Output packets: 0
    Security: Zone: HOST
    Allowed host-inbound traffic : any-service bfd bgp dvmrp igmp ldp msdp nhrp
    ospf pgm pim rip router-discovery rsvp sap vrrp
    Protocol inet, MTU: 1500
      Flags: Sendbcast-pkt-to-re
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 122.122.122/24, Local: 122.122.122.1,
        Broadcast: 122.122.122.255
    Protocol multiservice, MTU: Unlimited
      Flags: Is-Primary

  Logical interface et-4/0/0.32767 (Index 72) (SNMP ifIndex 517)
    Flags: SNMP-Traps 0x4004000 VLAN-Tag [ 0x0000.0 ]  Encapsulation: ENET2
    Input packets : 0
    Output packets: 0
    Security: Zone: HOST
    Allowed host-inbound traffic : any-service bfd bgp dvmrp igmp ldp msdp nhrp
    ospf pgm pim rip router-discovery rsvp sap vrrp
    Protocol multiservice, MTU: Unlimited
      Flags: None
```

The `Interface flags: Promiscuous` field shows that promiscuous mode is enabled on the interface.

- Verify that promiscuous mode works on the `et-4/0/0` interface. Send traffic into the `et-4/0/0` interface with a MAC address that is different from the interface MAC address and turn on promiscuous mode. From operational mode, enter the `monitor interface traffic` command.

```
user@host> monitor interface traffic
```

```
Interface    Link  Input packets       (pps)     Output packets      (pps)
gr-0/0/0      Up              0          (0)               0          (0)
ip-0/0/0      Up              0          (0)               0          (0)
lt-0/0/0      Up              0          (0)               0          (0)
xe-1/2/0    Down              0          (0)               0          (0)
xe-1/2/1    Down              0          (0)               0          (0)
xe-1/2/2    Down              0          (0)               0          (0)
xe-1/2/3    Down              0          (0)               0          (0)
```

```
xe-1/2/4    Down              0          (0)               0          (0)
xe-1/2/5    Down              0          (0)               0          (0)
xe-1/2/6    Down              0          (0)               0          (0)
xe-1/2/7    Down              0          (0)               0          (0)
xe-1/2/8    Down              0          (0)               0          (0)
xe-1/2/9    Down              0          (0)               0          (0)
et-4/0/0      Up        4403996     (100002)               0          (0)
et-4/2/0    Up                3          (0)         4403924      (99997)
avs0        Up                0          (0)               0          (0)
avs1        Up                0          (0)               0          (0)
dsc         Up                0                            0
em0         Up            15965                        14056
```

The `input packets` and `pps` fields show that traffic is passing through the `et-4/0/0` interface as expected after promiscuous mode is enabled.

- Verify that disabled promiscuous mode works on the `et-4/0/0` interface. Send traffic and turn off the promiscuous mode.

```
user@host> monitor interface traffic
```

```
Interface    Link  Input packets       (pps)     Output packets      (pps)
gr-0/0/0      Up               0          (0)               0          (0)
ip-0/0/0      Up               0          (0)               0          (0)
lt-0/0/0      Up               0          (0)               0          (0)
xe-1/2/0    Down               0          (0)               0          (0)
xe-1/2/1    Down               0          (0)               0          (0)
xe-1/2/2    Down               0          (0)               0          (0)
xe-1/2/3    Down               0          (0)               0          (0)
xe-1/2/4    Down               0          (0)               0          (0)
xe-1/2/5    Down               0          (0)               0          (0)
xe-1/2/6    Down               0          (0)               0          (0)
xe-1/2/7    Down               0          (0)               0          (0)
xe-1/2/8    Down               0          (0)               0          (0)
xe-1/2/9    Down               0          (0)               0          (0)
et-4/0/0      Up        11505495          (0)               0          (0)
et-4/2/0    Up                6          (0)         11505425          (0)
avs0        Up                0          (0)               0          (0)
avs1        Up                0          (0)               0          (0)
```

```
dsc             Up              0                       0
em0             Up          37964                   31739
```

The pps field shows that the traffic is not passing through the et-4/0/0 interface after promiscuous mode is disabled.

RELATED DOCUMENTATION

- Understanding Interfaces

Configuring Aggregated Ethernet Interfaces

IN THIS SECTION

The below topics discuss the overview Aggregated Ethernet (AE) interfaces on security devices, configuration details of AE interfaces, physical interfaces, AE interface link speed, VLAN tagging for aggregated Ethernet interfaces, and deleting an Aggregated Ethernet interface in security devices.

Understanding Aggregated Ethernet Interfaces

IN THIS SECTION

Link aggregation of Ethernet interfaces is defined in the IEEE 802.3ad standard. Junos OS implementation of 802.3ad balances traffic across the member links within an aggregated Ethernet bundle based on Layer 3 information carried in the packet, Layer 4 information carried in the packet, or both, or based on session ID data. (The session ID data has higher precedence than the Layer 3 or 4 information.) This implementation uses the same load-balancing algorithm used for per-packet load balancing.

Aggregated Ethernet interfaces can be Layer 3 interfaces (VLAN-tagged or untagged) and Layer 2 interfaces.

NOTE: This topic is specific to the SRX3000 and SRX5000 line devices. For information about link aggregation for other SRX Series Firewalls, see the "Configuring Link Aggregation Control Protocol" on page 152.

This topic contains the following sections:

LAGs

You can combine multiple physical Ethernet ports to form a logical point-to-point link, known as a link aggregation group (LAG) or bundle, such that a media access control (MAC) client can treat the LAG as if it were a single link. Support for LAGs based on IEEE 802.3ad makes it possible to aggregate physical interface links on your device. LAGs provide increased interface bandwidth and link availability by linking physical ports and load-balancing traffic crossing the combined interface. For the LAG to operate correctly, it is necessary to coordinate the two end systems connected by the LAG, either manually or automatically.

Internally, a LAG is a virtual interface presented on SRX3000 and SRX5000 line devices or on any system (consisting of devices such as routers and switches) supporting 802.3ad link aggregation. Externally, a LAG corresponds to a bundle of physical Ethernet links connected between an SRX3000 or SRX5000 line device and another system capable of link aggregation. This bundle of physical links is a virtual link.

Follow these guidelines for aggregated Ethernet support for the SRX3000 and SRX5000 lines:

- The devices support a maximum of 16 physical interfaces per single aggregated Ethernet bundle.
- Aggregated Ethernet interfaces can use interfaces from the same or different Flexible PIC Concentrators (FPCs) and PICs.
- On the aggregated bundle, capabilities such as MAC accounting, VLAN rewrites, and VLAN queuing are available.

LACP

Junos OS supports the Link Aggregation Control Protocol (LACP), which is a subcomponent of IEEE 802.3ad. LACP provides additional functionality for LAGs.

Starting with Junos OS Release 15.1X49-D40, LACP is supported on Layer 2 transparent mode in addition to existing support on Layer 3 mode. For information about link aggregation for other SRX Series Firewalls, see the Ethernet Switching User Guide.

LACP provides a standardized means for exchanging information between partner (remote or far-end of the link) systems on a link. This exchange allows their link aggregation control instances to reach agreement on the identity of the LAG to which the link belongs, and then to move the link to that LAG. This exchange also enables the transmission and reception processes for the link to function in an orderly manner.

For example, when LACP is not enabled, a local LAG might attempt to transmit packets to a remote individual interface, which causes the communication to fail. (An individual interface is a nonaggregatable interface.) When LACP is enabled, a local LAG cannot transmit packets unless a LAG with LACP is also configured on the remote end of the link.

You configure an aggregated Ethernet virtual link by specifying the link number as a physical device. Then you associate a set of ports that have the same speed and are in full-duplex mode. The physical ports can be 100-megabit Ethernet, 1-Gigabit Ethernet, and 10-Gigabit Ethernet.

When configuring LACP, follow these guidelines:

- LACP does not support automatic configuration on SRX3000 and SRX5000 line devices, but partner systems are allowed to perform automatic configuration. When an SRX3000 or SRX5000 line device is connected to a fully 802.3ad-compliant partner system, static configuration of LAGs is initiated on the SRX3000 and SRX5000 line device side, and static configuration is not needed on the partner side.
- When an SRX3000 or SRX5000 line device is connected to a Juniper Networks MX Series router, static configuration of LAGs is needed at both the actor (local or near-end of the link) and partner systems.

- Although the LACP functions on the SRX3000 and SRX5000 line devices are similar to the LACP features on Juniper Networks MX Series routers, the following LACP features on MX Series routers are not supported on SRX3000 and SRX5000 line devices: link protection, system priority, and port priority for aggregated Ethernet interfaces. Instead, SRX3000 and SRX5000 line devices provide active/standby support with redundant Ethernet interface LAGs in *chassis cluster* deployments.

LACP is supported in standalone deployments, where aggregated Ethernet interfaces are supported, and in chassis cluster deployments, where aggregated Ethernet interfaces and redundant Ethernet interfaces are supported simultaneously.

Configuring Aggregated Ethernet Interfaces

NOTE: This topic is specific to the SRX3000 and SRX5000 line devices.

To configure an aggregated Ethernet interface:

1. Set the number of aggregated Ethernet interfaces on the device. See Example: Configuring the Number of Aggregated Ethernet Interfaces on a Device.
2. Associate a physical interface with the aggregated Ethernet interface. See "Example: Associating Physical Interfaces with Aggregated Ethernet Interfaces" on page 140.
3. (Optional) Set the required link speed for all the interfaces included in the bundle. See "Example: Configuring Aggregated Ethernet Link Speed" on page 142.
4. (Optional) Configure the minimum number of links that must be up for the bundle as a whole to be labeled as up. See "Example: Configuring Aggregated Ethernet Minimum Links" on page 144.
5. (Optional) Enable or disable VLAN tagging. See "Understanding VLAN Tagging for Aggregated Ethernet Interfaces" on page 149.
6. (Optional) Enable promiscuous mode. See "Understanding Promiscuous Mode for Aggregated Ethernet Interfaces" on page 149.

SEE ALSO

Ethernet Switching User Guide

Understanding Physical Interfaces for Aggregated Ethernet Interfaces

You associate a physical interface with an aggregated Ethernet interface. Doing so associates the physical child links with the logical aggregated parent interface to form a link aggregation group (LAG). You must also specify the constituent physical links by including the `802.3ad` *configuration statement*.

A physical interface can be added to any aggregated Ethernet interface as long as all member links have the same link speed and the maximum number of member links does not exceed 16. The aggregated Ethernet interface instance number ae*x* can be from 0 through 127, for a total of 128 aggregated interfaces.

NOTE:

- If you specify (on purpose or accidentally) that a link already associated with an aggregated Ethernet interface be associated with another aggregated Ethernet interface, the link is removed from the previous interface (there is no need for you to explicitly delete it) and it is added to the other one.
- On SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550M devices, when you create an aggregated interface with two or more ports and if a link in the bundle goes down, the traffic forwarded through the same link will be rerouted two seconds later. This causes an outage for the traffic being sent to the link until reroute is complete.

Example: Associating Physical Interfaces with Aggregated Ethernet Interfaces

IN THIS SECTION

This example shows how to associate physical interfaces with aggregated Ethernet interfaces.

Requirements

Before you begin, set the number of aggregated Ethernet interfaces on the device. See Example: Configuring the Number of Aggregated Ethernet Interfaces on a Device.

Overview

In this example, you associate the physical child link of the ge-1/0/0 and ge-2/0/0 physical interfaces with the logical aggregate parent, ae0, thereby creating a LAG. Similarly, you create a LAG that associate the ge-3/0/0, ge-3/0/1, and ge-4/0/1 physical interfaces with the ae1 aggregated Ethernet interface.

Configuration

IN THIS SECTION

- Procedure | 141

Procedure

Step-by-Step Procedure

To associate physical interfaces with aggregated Ethernet interfaces:

1. Create the first LAG.

```
[edit]
user@host# set interfaces ge-1/0/0 gigether-options 802.3ad ae0
user@host# set interfaces ge-2/0/0 gigether-options 802.3ad ae0
```

2. Create the second LAG.

```
[edit]
user@host# set interfaces ge-3/0/0 gigether-options 802.3ad ae1
user@host# set interfaces ge-3/0/1 gigether-options 802.3ad ae1
user@host# set interfaces ge-4/0/0 gigether-options 802.3ad ae1
```

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

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

Verification

To verify the configuration is working properly, enter the `show interfaces` command.

Understanding Aggregated Ethernet Interface Link Speed

On aggregated Ethernet interfaces, you can set the required link speed for all interfaces included in the bundle. All interfaces that make up a bundle must be the same speed. If you include in the aggregated Ethernet interface an individual link that has a speed different from the speed you specify in the `link-speed` parameter, an error message will be logged.

The speed value is specified in bits per second either as a complete decimal number or as a decimal number followed by the abbreviation k (1000), m (1,000,000), or g (1,000,000,000).

Aggregated Ethernet interfaces on SRX3000 and SRX5000 line devices can have one of the following speed values:

- 100m—Links are 100 Mbps.
- 10g—Links are 10 Gbps.
- 1g—Links are 1 Gbps.

Example: Configuring Aggregated Ethernet Link Speed

IN THIS SECTION

This example shows how to configure the aggregated Ethernet link speed.

Requirements

Before you begin:

- Add the aggregated Ethernet interfaces using the device count. See Example: Configuring the Number of Aggregated Ethernet Interfaces on a Device.
- Associate physical interfaces with the aggregated Ethernet Interfaces. See "Example: Associating Physical Interfaces with Aggregated Ethernet Interfaces" on page 140.

Overview

In this example, you set the required link speed for all interfaces included in the bundle to 10 Gbps. All interfaces that make up a bundle must be the same speed.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To configure the aggregated Ethernet link speed:

1. Set the link speed.

```
[edit]
user@host# set interfaces ae0 aggregated-ether-options link-speed 10g
```

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

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

Verification

To verify the configuration is working properly, enter the `show interfaces` command.

Understanding Minimum Links for Aggregated Ethernet Interfaces

On aggregated Ethernet interfaces, you can configure the minimum number of links that must be up for the bundle as a whole to be labeled as up. By default, only one link must be up for the bundle to be labeled as up.

On SRX1000, SRX3000, and SRX5000 line devices, the valid range for the minimum links number is 1 through 16. When the maximum value (16) is specified, all configured links of a bundle must be up for the bundle to be labeled as up.

If the number of links configured in an aggregated Ethernet interface is less than the `minimum-links` value configured in the `minimum-links` statement, the configuration commit fails and an error message is displayed.

Example: Configuring Aggregated Ethernet Minimum Links

IN THIS SECTION

This example shows how to configure the minimum number of links on an aggregated Ethernet interface that must be up for the bundle as a whole to be labeled as up.

Requirements

Before you begin:

- Add the aggregated Ethernet interfaces using the device count. See Example: Configuring the Number of Aggregated Ethernet Interfaces on a Device.

- Associate physical interfaces with the aggregated Ethernet Interfaces. See "Example: Associating Physical Interfaces with Aggregated Ethernet Interfaces" on page 140.
- Configure the aggregated Ethernet link speed. See "Example: Configuring Aggregated Ethernet Link Speed" on page 142.

Overview

In this example, you specify that on interface ae0 at least eight links must be up for the bundle as a whole to be labeled as up.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To configure the minimum number of links on an aggregated Ethernet interface:

1. Set the minimum number of links.

```
[edit]
user@host# set interfaces ae0 aggregated-ether-options minimum-links 8
```

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

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

Verification

To verify the configuration is working properly, enter the `show interfaces` command.

Deleting Aggregated Ethernet Interface

You can delete an aggregated Ethernet interface from the interface configuration. Junos OS removes the configuration statements related to ae*x* and sets this interface to the down state. The deleted aggregated Ethernet interface still exists, but it becomes an empty interface.

Example: Deleting Aggregated Ethernet Interfaces

IN THIS SECTION

This example shows how to delete aggregated Ethernet interfaces using the device count.

Requirements

Before you begin, set the number of aggregated Ethernet interfaces on the device. See Example: Configuring the Number of Aggregated Ethernet Interfaces on a Device.

Overview

This example shows how to clean up unused aggregated Ethernet interfaces. In this example, you reduce the number of interfaces from 10 to 6, thereby removing the last 4 interfaces from the interface object list.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To delete an interface:

1. Set the number of aggregated Ethernet interfaces.

```
[edit]
user@host# delete chassis aggregated-devices ethernet device-count 6
```

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

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

Verification

To verify the configuration is working properly, enter the `show chassis aggregated-devices` command.

Example: Deleting Aggregated Ethernet Interface Contents

IN THIS SECTION

This example shows how to delete the contents of an aggregated Ethernet interface.

Requirements

Before you begin:

- Set the number of aggregated Ethernet interfaces on the device. See Example: Configuring the Number of Aggregated Ethernet Interfaces on a Device.
- Associate a physical interface with the aggregated Ethernet interface. See "Example: Associating Physical Interfaces with Aggregated Ethernet Interfaces" on page 140.
- Set the required link speed for all the interfaces included in the bundle. See "Example: Configuring Aggregated Ethernet Link Speed" on page 142.
- Configure the minimum number of links that must be up for the bundle as a whole to be labeled as up. See "Example: Configuring Aggregated Ethernet Minimum Links" on page 144.

Overview

In this example, you delete the contents of the ae4 aggregated Ethernet interface, which sets it to the down state.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To delete the contents of an aggregated Ethernet interface:

1. Delete the interface.

```
[edit]
user@host# delete interfaces ae4
```

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

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

Verification

To verify the configuration is working properly, enter the `show interfaces` command.

Understanding VLAN Tagging for Aggregated Ethernet Interfaces

Aggregated Ethernet interfaces can be either VLAN-tagged or untagged, with LACP enabled or disabled. Aggregated Ethernet interfaces on the SRX3000 and SRX5000 lines support the configuration of `native-vlan-id`, which consists of the following configuration statements:

- `inner-tag-protocol-id`
- `inner-vlan-id`
- `pop-pop`
- `pop-swap`
- `push-push`
- `swap-push`
- `swap-swap`

Understanding Promiscuous Mode for Aggregated Ethernet Interfaces

You can enable promiscuous mode on aggregated Ethernet interfaces. When promiscuous mode is enabled on a Layer 3 Ethernet interface, all packets received on the interface are sent to the central point or Services Processing Unit (SPU) regardless of the destination MAC address of the packet. If you enable promiscuous mode on an aggregated Ethernet interface, promiscuous mode is then enabled on all member interfaces.

Verifying Aggregated Ethernet Interfaces

IN THIS SECTION

Verifying Aggregated Ethernet Interfaces (terse)

IN THIS SECTION

Purpose

Display status information in terse (concise) format for aggregated Ethernet interfaces.

Action

From operational mode, enter the `show interfaces ae0 terse` command.

```
user@host> show interfaces ae0 terse
ge-2/0/0.0              up    up   aenet    --> ae0.0
ge-2/0/0.32767          up    up   aenet    --> ae0.32767
ge-2/0/1.0              up    up   aenet    --> ae0.0
ge-2/0/1.32767          up    up   aenet    --> ae0.32767
ae0                     up    up
ae0.0                   up    up   bridge
ae0.32767               up    up   multiservice
```

The output shows the bundle relationship for the aggregated Ethernet interface and the overall status of the interface, including the following information:

- The link aggregation control PDUs run on the .0 child logical interfaces for the untagged aggregated Ethernet interface.
- The link aggregation control PDUs run on the .32767 child logical interfaces for the VLAN-tagged aggregated Ethernet interface.
- The .32767 logical interface is created for the parent link and all child links.

Verifying Aggregated Ethernet Interfaces (extensive)

IN THIS SECTION

Purpose

Display status information and statistics in extensive (detailed) format for aggregated Ethernet interfaces.

Action

From operational mode, enter the `show interfaces ae0 extensive` command.

```
user@host> show interfaces ae0 extensive
Physical interface: ae0, Enabled, Physical link is Up
  ...
  Logical interface ae0.0 (Index 67) (SNMP ifIndex 628) (Generation 134)
    ...
    LACP info:        Role     System            System      Port    Port  Port
                             priority          identifier  priority  number   key
      ge-5/0/0.0     Actor        127  00:1f:12:8c:af:c0       127     832     1
      ge-5/0/0.0   Partner        127  00:1f:12:8f:d7:c0       127     640     1
      ge-5/0/1.0     Actor        127  00:1f:12:8c:af:c0       127     833     1
      ge-5/0/1.0   Partner        127  00:1f:12:8f:d7:c0       127     641     1
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-5/0/0.0             12830        7090            0            0
      ge-5/0/1.0             10304        4786            0            0
    ...
  Logical interface ae0.32767 (Index 70) (SNMP ifIndex 630) (Generation 135)
    ...
    LACP info:        Role     System            System      Port    Port  Port
                             priority          identifier  priority  number   key
      ge-5/0/0.32767    Actor      127 00:1f:12:8c:af:c0       127     832     1
      ge-5/0/0.32767  Partner      127 00:1f:12:8f:d7:c0       127     640     1
      ge-5/0/1.32767    Actor      127 00:1f:12:8c:af:c0       127     833     1
      ge-5/0/1.32767  Partner      127 00:1f:12:8f:d7:c0       127     641     1
```

```
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-5/0/0.32767         12830        7090            0            0
      ge-5/0/1.32767         10304        4786            0            0
  ...
```

The output shows detailed aggregated Ethernet interface information. This portion of the output shows LACP information and LACP statistics for each logical aggregated Ethernet interface.

RELATED DOCUMENTATION

Configuring Aggregated Ethernet Interfaces | 139

Change History Table

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

Release	Description
15.1X49-D40	Starting with Junos OS Release 15.1X49-D40, LACP is supported on Layer 2 transparent mode in addition to existing support on Layer 3 mode.

Configuring Link Aggregation Control Protocol

IN THIS SECTION

Link Aggregation Control Protocol (LACP) provides a standard means for information exchange between the systems on a link. The below topics discuss the overview of LACP on standalone devices, examples of configuring LACP, LAG and LACP support line devices.

Understanding LACP on Standalone Devices

Link Aggregation Control Protocol (LACP) provides a standardized means for exchanging information between partner systems on a link. Within LACP, the local end of a child link is known as the actor and the remote end of the link is known as the partner.

LACP is enabled on an aggregated Ethernet interface by setting the mode to either passive or active. However, to initiate the transmission of link aggregation control protocol data units (PDUs) and response link aggregation control PDUs, you must enable LACP at both the local and remote ends of the links, and one end must be active:

- Active mode—If either the actor or partner is active, they exchange link aggregation control PDUs. The actor sends link aggregation control PDUs to its protocol partner that convey what the actor knows about its own state and that of the partner's state.
- Passive mode—If the actor and partner are both in passive mode, they do not exchange link aggregation control PDUs. As a result, the aggregated Ethernet links do not come up. In passive transmission mode, links send out link aggregation control PDUs only when they receive them from the remote end of the same link.

By default, the actor and partner transmit link aggregation control PDUs every second. You can configure different periodic rates on active and passive interfaces. When you configure the actor and partner interfaces at different rates, the transmitter (actor) honors the receiver's (partner's) rate.

You configure the interval at which the interfaces on the remote side of the link transmit link aggregation control PDUs by configuring the `periodic` statement on the interfaces on the local side. It is the configuration on the local side that specifies the behavior of the remote side. That is, the remote side transmits link aggregation control PDUs at the specified interval. The interval can be `fast` (every second) or `slow` (every 30 seconds).

NOTE: Starting with Junos OS Release 15.1X49-D40, LACP is supported in Layer 2 transparent mode in addition to existing support in Layer 3 mode.

Example: Configuring Link Aggregation Control Protocol

IN THIS SECTION

This example shows how to configure LACP.

Requirements

This example uses an SRX Series Firewall.

Before you begin:

- Determine which interfaces to use and verify that they are in switch mode. See Understanding VLANs.

Overview

In this example, for aggregated Ethernet interfaces, you configure the Link Aggregation Control Protocol (LACP). LACP is one method of bundling several physical interfaces to form one logical interface.

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

```
set interfaces ge-0/0/6 ether-options 802.3ad ae0
set interfaces ge-0/0/7 ether-options 802.3ad ae0
```

```
set interfaces ae0 vlan-tagging
set interfaces ae0 aggregated-ether-options lacp active periodic fast
set interfaces ae0 unit 0 family ethernet-switching interface-mode trunk
set vlan vlan1000 vlan-id 1000
set interfaces ae0 unit 0 family ethernet-switching vlan members vlan1000
```

Step-by-Step Procedure

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

To configure LACP:

1. Configure the interfaces for ae0.

```
[edit ]
user@host# set interfaces ge-0/0/6 ether-options 802.3ad ae0
user@host# set interfaces ge-0/0/7 ether-options 802.3ad ae0
```

2. Configure ae0 interface for vlan tagging.

```
[edit ]
user@host# set interfaces ae0 vlan-tagging
```

3. Configure LACP for ae0 and configure periodic transmission of LACP packets.

```
[edit ]
user@host# set interfaces ae0 aggregated-ether-options lacp active periodic fast
```

4. Configure ae0 as a trunk port.

```
[edit ]
user@host# set interfaces ae0 unit 0 family ethernet-switching interface-mode trunk
```

5. Configure the VLAN.

```
[edit ]
user@host# set vlan vlan1000 vlan-id 1000
```

6. Add the ae0 interface to the VLAN.

```
[edit ]
user@host# set interfaces ae0 unit 0 family ethernet-switching vlan members vlan1000
```

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

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

Results

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

```
[edit]
user@host# show interfaces
ge-0/0/6 {
    ether-options {
        802.3ad ae0;
    }
}
ge-0/0/7 {
    ether-options {
        802.3ad ae0;
    }
}
ae0 {
    vlan- tagging;
    aggregated-ether-options {
        lacp {
            active;
            periodic fast;
```

```
        }
    }
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members vlan1000;
            }
        }
    }
}
```

Verification

IN THIS SECTION

Verifying LACP Statistics

Purpose

Display LACP statistics for aggregated Ethernet interfaces.

Action

From operational mode, enter the `show lacp statistics interfaces ae0` command.

```
user@host> show lacp statistics interfaces ae0
Aggregated interface: ae0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-0/0/6                1352        2035            0           0
      ge-0/0/7                1352        2056            0           0
```

Meaning

The output shows LACP statistics for each physical interface associated with the aggregated Ethernet interface, such as the following:

- The LACP received counter that increments for each normal hello packet received
- The number of LACP transmit packet errors logged
- The number of unrecognized packet errors logged
- The number of invalid packets received

Use the following command to clear the statistics and see only new changes:

```
user@host# clear lacp statistics interfaces ae0
```

Verifying LACP Aggregated Ethernet Interfaces

Purpose

Display LACP status information for aggregated Ethernet interfaces.

Action

From operational mode, enter the `show lacp interfaces ae0` command.

```
user@host> show lacp interfaces ae0
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity
      ge-0/0/6       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-0/0/6     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Passive
      ge-0/0/7       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-0/0/7     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Passive
    LACP protocol:        Receive State  Transmit State          Mux State
      ge-0/0/6                  Current   Fast periodic Collecting distributing
      ge-0/0/7                  Current   Fast periodic Collecting distributing
```

Meaning

The output shows aggregated Ethernet interface information, including the following information:

- The LACP state—Indicates whether the link in the bundle is an actor (local or near-end of the link) or a partner (remote or far-end of the link).
- The LACP mode—Indicates whether both ends of the aggregated Ethernet interface are enabled (active or passive)—at least one end of the bundle must be active.
- The periodic link aggregation control PDU transmit rate.
- The LACP protocol state—Indicates the link is up if it is collecting and distributing packets.

SEE ALSO

Understanding Link Aggregation Control Protocol

Ethernet Ports Switching Overview for Security Devices

Verifying LACP on Standalone Devices

IN THIS SECTION

Verifying LACP Statistics

IN THIS SECTION

Purpose

Display LACP statistics for aggregated Ethernet interfaces.

Action

From operational mode, enter the `show lacp statistics interfaces ae0` command.

```
user@host> show lacp statistics interfaces ae0
Aggregated interface: ae0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-2/0/0                1352        2035            0            0
      ge-2/0/1                1352        2056            0            0
      ge-2/2/0                1352        2045            0            0
      ge-2/2/1                1352        2043            0            0
```

The output shows LACP statistics for each physical interface associated with the aggregated Ethernet interface, such as the following:

- The LACP received counter that increments for each normal hello
- The number of LACP transmit packet errors logged
- The number of unrecognized packet errors logged
- The number of invalid packets received

Use the following command to clear the statistics and see only new changes:

```
user@host# clear lacp statistics interfaces ae0
```

Verifying LACP Aggregated Ethernet Interfaces

IN THIS SECTION

Purpose

Display LACP status information for aggregated Ethernet interfaces.

Action

From operational mode, enter the `show lacp interfaces ae0` command.

```
user@host> show lacp interfaces ae0
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity
      ge-2/0/0       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/0/0     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/0/1       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/0/1     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/2/0       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/2/0     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/2/1       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/2/1     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
    LACP protocol:        Receive State  Transmit State          Mux State
      ge-2/0/0                  Current   Fast periodic Collecting distributing
      ge-2/0/1                  Current   Fast periodic Collecting distributing
      ge-2/2/0                  Current   Fast periodic Collecting distributing
      ge-2/2/1                  Current   Fast periodic Collecting distributing
```

The output shows aggregated Ethernet interface information, including the following information:

- The LACP state—Indicates whether the link in the bundle is an actor (local or near-end of the link) or a partner (remote or far-end of the link).
- The LACP mode—Indicates whether both ends of the aggregated Ethernet interface are enabled (active or passive)—at least one end of the bundle must be active.
- The periodic link aggregation control PDU transmit rate.
- The LACP protocol state—Indicates the link is up ifit is collecting and distributing packets.

RELATED DOCUMENTATION

Verifying LACP on Redundant Ethernet Interfaces

LAG and LACP Support Line Devices with I/O Cards (IOCs)

IN THIS SECTION

NOTE: The following notes apply to 'LAG and LACP Support on SRX5000 Line Devices' as outlined in this document.

- Cross-IOC LAG interfaces do not support Layer 2 transparent mode.
- Mixed interface speeds are supported on the same aggregated bundle.
- A redundant Ethernet interface or aggregated Ethernet interface must contain child interfaces from the same IOC type.

LAG and LACP Support on the SRX5000 Module Port Concentrator

The SRX5000 Module Port Concentrator (SRX5K-MPC) on SRX5400, SRX5600, and SRX5800 devices supports link aggregation groups (LAGs) and Link Aggregation Control Protocol (LACP).

Support for LAGs based on IEEE 802.3ad makes it possible to aggregate physical interface links on your device. LAGs provide increased interface bandwidth and link availability by linking physical ports and load-balancing traffic crossing the combined interface.

LACP provides a standardized means for exchanging information between partner (remote or far-end of the link) systems on a link. This exchange allows their link aggregation control instances to reach agreement on the identity of the LAG to which the link belongs, and then to move the link to that LAG. This exchange also enables the transmission and reception processes for the link to function in an orderly manner.

The following LAG and LACP features are supported on the SRX5K-MPC:

- Bandwidth aggregation—Increases bandwidth, provides graceful degradation as failure occurs, and increases availability.
- Link redundancy and load balancing (within chassis cluster)—Provides network redundancy by load-balancing traffic across all available links. If one of the links should fail, the system automatically load-balances traffic across all remaining links.

- Dynamic link management—Enables automatic addition and deletion of individual links to the aggregate bundle without user intervention.

LACP supports the following features:

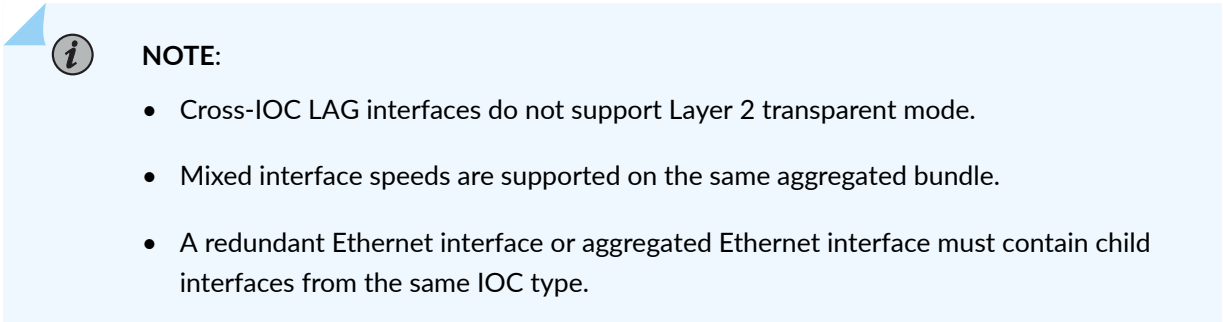
- LACP bundles several physical interfaces to form one logical interface by exchanging LACP packets between the local interface and the remote interface. LACP monitors the link for changes in interface state by exchanging a periodic LACP heartbeat between two sides. Any changes in interface state are reflected in the LACP packet.
- Normally after an LACP is configured and committed, two sides start to exchange interface and port information. Once they identify each other and match the LACP state machine criteria, the LACP is declared as up. You can deactivate or delete the LACP configuration.
- By default, the LACP packets are exchanged in every second. You can configure the LACP interval as fast (every second) or slow (every 30 seconds) to ensure the health of the interfaces.
- LACP supports distributed and centralized modes. Chassis cluster setup is recommended to operate with LACP distributed mode, which handles chassis cluster failover better. The centralized mode might experience traffic loss during failover.

SRX5K-MPCs on SRX5000 line devices provide active and standby support with redundant Ethernet interface LAGs in chassis cluster deployments.

LAG and LACP Support on the SRX5000 Line IOCs in Express Path Mode

Starting in Junos OS Release 15.1X49-D40, the IOC2 and IOC3 cards on SRX5400, SRX5600, and SRX5800 devices support link aggregation groups (LAGs) and Link Aggregation Control Protocol (LACP) in Express Path mode.

You can use the links in a LAG as ingress or egress interfaces in Express Path mode. The LAG links can include links from cards such as IOC2 or IOC3. For a LAG link to qualify for Express Path, all its member links should be connected to Express Path-enabled network processors. If Express Path is disabled on any of the member links in a LAG, a regular session (non-Express Path session) is created.



NOTE:

- Cross-IOC LAG interfaces do not support Layer 2 transparent mode.
- Mixed interface speeds are supported on the same aggregated bundle.
- A redundant Ethernet interface or aggregated Ethernet interface must contain child interfaces from the same IOC type.

SEE ALSO

Configuring Aggregated Ethernet Interfaces | 139

Configuring Link Aggregation Control Protocol | 152

Example: Configuring LAG Interface on an Line Device with IOC2 or IOC3

IN THIS SECTION

Starting in Junos OS Release 15.15X49-D40, IEEE 802.3ad link aggregation enables you to group Ethernet interfaces to form a single, aggregated Ethernet interface. This single, aggregated Ethernet interface is also known as a LAG or bundle. The LACP provides additional functionality for LAGs.

This example shows how to configure LAG on an SRX Series Firewall using the links from either IOC2 or IOC3 in Express Path mode.

Requirements

This example uses the following software and hardware components:

- Junos OS Release 15.1X49-D40 or later for SRX Series Firewalls.
- SRX5800 with IOC2 or IOC3 with Express Path enabled on IOC2 and IOC3. For details, see *Express Path*.

Overview

In this example, you create a logical aggregated Ethernet interface and define the parameters associated with the logical aggregated Ethernet interface, such as a logical unit, interface properties, and LACP. Next, define the member links to be contained within the aggregated Ethernet interface—for example, four 10-Gigabit Ethernet interfaces. Finally, configure an LACP for link detection.

The following member links are used in this example:

- xe-0/0/8

- xe-0/0/9
- xe-1/0/8
- xe-1/0/9
- xe-3/1/4
- xe-3/1/5
- xe-5/1/4
- xe-5/1/5

Configuration

IN THIS SECTION

CLI Quick Configuration

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

```
set chassis aggregated-devices ethernet device-count 5
set interfaces xe-0/0/8 gigether-options 802.3ad ae1
set interfaces xe-0/0/9 gigether-options 802.3ad ae0
set interfaces xe-1/0/8 gigether-options 802.3ad ae1
set interfaces xe-1/0/9 gigether-options 802.3ad ae0
set interfaces xe-3/1/4 gigether-options 802.3ad ae1
set interfaces xe-3/1/5 gigether-options 802.3ad ae0
set interfaces xe-5/1/4 gigether-options 802.3ad ae1
set interfaces xe-5/1/5 gigether-options 802.3ad ae0
set interfaces ae0 unit 0 family inet address 17.0.0.1/24
set interfaces ae1 unit 0 family inet address 16.0.0.1/24
```

```
set interfaces ae0 aggregated-ether-options lacp active
set interfaces ae1 aggregated-ether-options lacp active
```

Procedure

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see the Junos OS CLI User Guide.

To configure LAG Interfaces:

1. Specify the number of aggregated Ethernet interfaces to be created.

```
[edit chassis]
user@host# set aggregated-devices ethernet device-count 5
```

2. Specify the members to be included within the aggregated Ethernet bundle.

```
[edit interfaces]
user@host# set xe-0/0/8 gigether-options 802.3ad ae1
user@host# set xe-0/0/9 gigether-options 802.3ad ae0
user@host# set xe-1/0/8 gigether-options 802.3ad ae1
user@host# set xe-1/0/9 gigether-options 802.3ad ae0
user@host# set xe-3/1/4 gigether-options 802.3ad ae1
user@host# set xe-3/1/5 gigether-options 802.3ad ae0
user@host# set xe-5/1/4 gigether-options 802.3ad ae1
user@host# set xe-5/1/5 gigether-options 802.3ad ae0
```

3. Assign an IP address to ae0 and ae1.

```
[edit interfaces]
user@host# set ae0 unit 0 family inet address 17.0.0.1/24
user@host# set ae1 unit 0 family inet address 16.0.0.1/24
```

4. Set the LACP on reth0.

```
[edit interfaces]
user@host# set ae0 aggregated-ether-options lacp active
user@host# set ae1 aggregated-ether-options lacp active
```

Results

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

```
[edit]
user@host# show interfaces
xe-0/0/8 {
    gigether-options {
        802.3ad ae1;
    }
}
xe-0/0/9 {
    gigether-options {
        802.3ad ae0;
    }
}
xe-1/0/8 {
    gigether-options {
        802.3ad ae1;
    }
}
xe-1/0/9 {
    gigether-options {
        802.3ad ae0;
    }
}
xe-3/1/4 {
    gigether-options {
        802.3ad ae1;
    }
}
xe-3/1/5 {
    gigether-options {
```

```
        802.3ad ae0;
    }
}
ae0 {
    aggregated-ether-options {
        lacp {
            active;
        }
    }
    unit 0 {
        family inet {
            address 17.0.0.1/24;
        }
    }
}
ae1 {
    aggregated-ether-options {
        lacp {
            active;
        }
    }
    unit 0 {
        family inet {
            address 16.0.0.1/24;
        }
    }
}
```

```
[edit]
user@host# show chassis
aggregated-devices {
    ethernet {
        device-count 5;
    }
}
```

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

Verification

IN THIS SECTION

Verifying LACP on Redundant Ethernet Interfaces

Purpose

Display LACP status information for redundant Ethernet interfaces.

Action

From operational mode, enter the `show lacp interfaces` command to check that LACP has been enabled as active on one end.

```
user@host> show lacp interfaces
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity
      xe-0/0/9       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      xe-0/0/9     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      xe-1/0/9       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      xe-1/0/9     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      xe-3/1/5       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      xe-3/1/5     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      xe-5/1/5       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      xe-5/1/5     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
    LACP protocol:        Receive State  Transmit State          Mux State
      xe-0/0/9                  Current   Fast periodic Collecting distributing
      xe-1/0/9                  Current   Fast periodic Collecting distributing
      xe-3/1/5                  Current   Fast periodic Collecting distributing
      xe-5/1/5                  Current   Fast periodic Collecting distributing

Aggregated interface: ae1
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity
      xe-0/0/8       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      xe-0/0/8     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      xe-1/0/8       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
```

```
    xe-1/0/8     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
    xe-3/1/4       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
    xe-3/1/4     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
    xe-5/1/4       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
    xe-5/1/4     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
  LACP protocol:        Receive State  Transmit State          Mux State
    xe-0/0/8                  Current   Fast periodic Collecting distributing
    xe-1/0/8                  Current   Fast periodic Collecting distributing
    xe-3/1/4                  Current   Fast periodic Collecting distributing
    xe-5/1/4                  Current   Fast periodic Collecting distributing
```

The output indicates that LACP has been set up correctly and is active at one end.

Change History Table

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

Release	Description
15.1X49-D40	Starting in Junos OS Release 15.1X49-D40, the IOC2 and IOC3 cards on SRX5400, SRX5600, and SRX5800 devices support link aggregation groups (LAGs) and Link Aggregation Control Protocol (LACP) in Express Path mode.
15.1X49-D40	Starting in Junos OS Release 15.15X49-D40, IEEE 802.3ad link aggregation enables you to group Ethernet interfaces to form a single, aggregated Ethernet interface.

Configuring Gigabit Ethernet Physical Interface Modules

IN THIS SECTION

Small form-factor pluggables (SFPs) are hot-pluggable modular interface transceivers for Gigabit connections. The 1-Port Gigabit Ethernet SFP Mini-PIM interfaces a single Gigabit Ethernet device or a network. The below topics discuss the overview and configuration of 1-Port Gigabit Ethernet SFP Mini-PIM interface, overview and configuration of 2-Port 10-GE XPIM and overview and configuration of 8-Port GE SFP XPIMs.

Understanding the 1-Port Gigabit Ethernet SFP Mini-PIM

IN THIS SECTION

Small form-factor pluggables (SFPs) are hot-pluggable modular interface transceivers for Gigabit connections. Gigabit Ethernet SFP Mini-PIMs can be used in copper and optical environments to provide maximum flexibility when upgrading from an existing infrastructure to Metro Ethernet.

The 1-Port Gigabit Ethernet SFP Mini-PIM interfaces a single Gigabit Ethernet device or a network. It supports a variety of transceivers with data speeds of 10-Mbps/100-Mbps/1-Gbps with extended LAN or WAN connectivity.

Transceivers are hot-swappable.

This topic includes the following sections:

Supported Features

The following features are supported on the 1-Port Gigabit Ethernet SFP Mini-PIM:

- 10-Mbps/100-Mbps/1-Gbps link speed

- Half-duplex/full-duplex support
- Autonegotiation
- Encapsulations
- Maximum transmission unit (MTU) size of 1514 bytes (default) and 9010 bytes (jumbo frames)
- Loopback
- Transceivers are hot-swappable

Interface Names and Settings

The following format is used to represent the 1-Port Gigabit Ethernet SFP Mini-PIM interface names:

type-fpc/pic/port

Where:

- type—Media type (ge)
- fpc—Number of the Flexible PIC Concentrator (FPC) card on which the physical interface is located
- pic—Number of the PIC on which the physical interface is located (0)
- port—Specific port on a PIC (0)

Examples: `ge-1/0/0` and `ge-2/0/0`

By default, the interfaces on the ports on the uplink module installed on the device are enabled. You can also specify the MTU size for the Gigabit Ethernet interface. Junos OS supports values from 256 through 9010. The default MTU size for Gigabit Ethernet interfaces is 1514.

Available Link Speeds and Modes

The 1-Port Gigabit Ethernet SFP Mini-PIM supports the following link speeds:

- `10m`—Sets the link speed to 10 Mbps.
- `100m`—Sets the link speed to 100 Mbps.
- `1g`—Sets the link speed to 1 Gbps.

The 1-Port Gigabit Ethernet SFP Mini-PIM supports the following link modes:

- `Full-duplex`—Allows bidirectional communication at a given point in time.
- `Half-duplex`—Allows single directional communication at a given point in time.

Link Settings

The 1-Port Gigabit Ethernet SFP Mini-PIM includes the following link settings:

- `auto-negotiation`—Enables autonegotiation of link mode and speed.

  NOTE: By default, autonegotiation is enabled. To disable autonegotiation, use `set gigether-options no-autonegotiation`
  We recommend enabling autonegotiation.

- `loopback`—Enables loopback.
- `no-auto-negotiation`—Disables autonegotiation of link mode and speed.
- `no-loopback`—Disables loopback.

By default a link speed of 1 Gbps in full-duplex mode is supported.

NOTE: On SRX340 High Memory devices, traffic might stop between the SRX340 device and the Cisco switch due to link mode mismatch. We recommend setting the same value to the autonegotiation parameters on both ends.

NOTE: On SRX300 devices, the link goes down when you upgrade FPGA on 1-Port Gigabit Ethernet SFP mini-PIM. As a workaround, run the `restart fpc` command and restart the FPC.

Example: Configuring the 1-Port Gigabit Ethernet SFP Mini-PIM Interface

IN THIS SECTION

This example shows how to perform basic configuration for the 1-Port Gigabit Ethernet SFP Mini-PIM.

Requirements

Before you begin:

- Establish basic connectivity. See the *Getting Started Guide* for your device.
- Configure network interfaces as necessary. See "Example: Creating an Ethernet Interface" on page 130.

Overview

In this example, you configure the ge-2/0/0 interface, set the operating speed to 100 Mbps, and define a logical interface that you can connect to the 1-Port Gigabit Ethernet SFP Mini-PIM. You also set the MTU value to 9010 and set the link option to no-loopback.

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

```
set interfaces ge-2/0/0 link-mode full-duplex speed 100m
set interface ge-2/0/0 gigether-options no-loopback
```

Configuring Physical Properties

GUI Quick Configuration

Step-by-Step Procedure

To quickly configure the physical properties of a 1-Port Gigabit Ethernet SFP Mini-PIM using J-Web, use the following steps:

1. Select `Configure > Interfaces`.
2. Under Interface, select `ge-2/0/0` and then click `Edit`. A pop-up window appears.
3. In the Description box, type the description for the SFP Mini-PIM.
4. In the MTU box, type `9010`.
5. From the Speed list, select `100Mbps`.
6. From the Link-mode list, select `Full-duplex`.
7. Select the Enable Auto-negotiation checkbox.
8. Select the Enable Per Unit Scheduler checkbox.
9. Click `OK`

Disabling the Interface

GUI Quick Configuration

Step-by-Step Procedure

To disable the 1-Port Gigabit Ethernet SFP Mini-PIM using J-Web, use the following steps:

1. Select `Configure > Interfaces` .
2. Under Interface, select `ge-2/0/0` and then click `Disable`.

Configuring Logical Properties

GUI Quick Configuration

Step-by-Step Procedure

To quickly configure the logical properties of a 1-Port Gigabit Ethernet SFP Mini-PIM using J-Web, use the following steps:

1. Select `Configure > Interfaces`.
2. Under Interface, select `ge-2/0/0.0`, and then click `Add Logical Interface`. A pop-up window appears.
3. In the Unit box, type `0`.
4. In the Description box, type a description for the SFP Mini-PIM.
5. From the Zone list, select `untrust`.
6. To edit the family protocol type to the Mini-PIM interfaces, select the IPv4 tab, and then select `Enable address configuration`.
7. Click `Add`, and then type IPv4 address.
8. Click `OK`.

Editing Logical Properties

Step-by-Step Procedure

To quickly configure the physical properties of a 1-Port Gigabit Ethernet SFP Mini-PIM using J-Web:

1. Under Interface, select the logical interface added to the 1-Port Gigabit Ethernet SFP Mini-PIM and then click `Edit`. A pop-up window appears.
2. Under Interface, select `ge-2/0/0.0`, and then click `Edit Logical Interface`. A pop-up window appears.
3. From the Zone list, select `trust`.
4. To enable DHCP client on the interface, select the IPv4 tab and then select `Enable DHCP`.
5. Click `OK`.

NOTE: You cannot add or edit Description and Unit for a logical interface.

Deleting the Logical Interface

GUI Quick Configuration

Step-by-Step Procedure

To delete the logical interface of 1–Port Gigabit Ethernet SFP Mini-PIM using J-Web,

1. Select `Configure > Interfaces`.
2. Under Interface, select `ge-2/0/0.0`, and then click `Delete`.

Configuring a 1-Port Gigabit Ethernet SFP Mini-PIM

Step-by-Step Procedure

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

To configure a 1-Port Gigabit Ethernet SFP Mini-PIM:

1. Configure the interface.

```
[edit]
user@host# edit interfaces ge-2/0/0
```

2. Set the operating link-mode full-duplex speed of 100 Mbps for the SFP Mini-PIM.

```
[edit interfaces ge-2/0/0]
user@host# set link-mode full-duplex speed 100m
```

3. Assign the MTU value.

```
[edit interfaces ge-2/0/0]
user@host# set mtu 9010
```

4. Add the logical interface.

```
[edit interfaces ge-2/0/0]
user@host# set unit 0 family inet address 14.1.1.1/24
```

5. Set the link options.

```
[edit interfaces ge-2/0/0]
user@host# set gigether-options no-loopback
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces ge-2/0/0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interfaces ge-2/0/0
    mtu 9010;
    speed 100m;
    gigether-options {
    no-loopback;
}
unit 0 {
    family inet {
    14.1.1.1/24
```

```
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying That the Correct Hardware Is Installed

Purpose

Verify that the 1-Port Gigabit Ethernet SFP Mini-PIM is installed on the device.

Action

From operational mode, enter the `show chassis hardware` command.

```
user@host> show chassis hardware detail
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                AG0309AA0004      SRX240b
Routing Engine   REV 16   750-021792   VL3180            RE-SRX240B
  da0     999 MB  ST72682                                Nand Flash
  usb0 (addr 1)  DWC OTG root hub 0    vendor 0x0000     uhub0
  usb0 (addr 2)  product 0x005a 90  vendor 0x0409     uhub1
  usb0 (addr 3)  ST72682  High Speed Mode 64218 STMicroelectronics umass0
FPC 0                                                  FPC
  PIC 0                                                16x GE Base PIC
FPC 1                    750-023367   112009000278      FPC
  PIC 0                                                1x T1E1 mPIM
```

```
FPC 2            REV 00   750-03273   AABC5081          FPC
  PIC 0                                                  1x GE High-Perf SFP mPIM
    Xcvr 0       REV 02   740-011612   9101465           SFP-T
FPC 4                     750-029145   122009000061      FPC
PIC 0                                                    1x GE SFP mPIM
    Xcvr 0       REV 01   740-011782   PBL0C3T           SFP-SX
Power Supply 0
```

Verify that the output contains the following values:

- `FPC 2, PIC 0` —1x GE High-Perf SFP mPIM
- `FPC 4, PIC 0` —1x GE SFP mPIM

NOTE: In the example shown above, the output for 1-Port SFP Mini-Physical Interface Module is displayed as 1X GE SFP mPIM and the output for 1-Port Gigabit Ethernet SFP Mini-Physical Interface Module is displayed as 1X GE High-Perf SFP mPIM.

NOTE: The 1-Port GE SFP Mini-PIM is installed in the second slot of the device chassis; therefore the output displayed is 1x GE High-Perf SFP mPIM and the Flexible PIC Concentrator (FPC) used here is fpc 2.

The 1-Port SFP Mini-PIM is installed in the fourth slot of the device chassis; therefore the output displayed is 1x GE SFP mPIM and Flexible PIC Concentrator (FPC) used here is fpc 4.

Verifying the FPC Status

Purpose

Verify the FPC status.

Action

From operational mode, enter the `show chassis fpc` command.

```
show@host> show chassis fpc
                     Temp  CPU Utilization (%)   Memory    Utilization (%)
Slot State           (C)  Total  Interrupt      DRAM (MB) Heap     Buffer
```

```
  0  Online          -------------------- CPU less FPC --------------------
  1  Online          -------------------- CPU less FPC --------------------
  2  Online          -------------------- CPU less FPC --------------------
  3  Empty
  4  Online          -------------------- CPU less FPC --------------------
```

The output should show the FPC status as online.

The 1-Port SFP Mini-PIM is installed in the fourth slot of the device chassis; the output shows the FPC status for slot 4 as online.

The 1-Port Gigabit Ethernet SFP Mini-PIM is installed in the second slot of the device chassis; the output shows the FPC status for slot 2 as online.

Verifying the Interface Settings

Purpose

Verify that the interface is configured as expected.

Action

From operational mode, enter the `show interface ge-2/0/0` command.

```
user@host# run show interfaces ge-2/0/0
Physical interface: ge-2/0/0, Enabled, Physical link is Up
  Interface index: 156, SNMP ifIndex: 552
  Link-level type: Ethernet, MTU: 9010, Link-mode: Full-duplex, Speed: 100mbps, BPDU Error:
None, MAC-REWRITE Error: None,
  Loopback: Disabled, Source filtering: Disabled, Flow control: Enabled, Auto-negotiation:
Enabled, Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:22:83:99:ac:f2, Hardware address: 00:22:83:99:ac:f2
  Last flapped   : 2010-08-17 12:20:33 UTC (00:00:20 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : None
  Active defects : None
```

```
Logical interface ge-2/0/0.0 (Index 88) (SNMP ifIndex 557)
  Flags: SNMP-Traps Encapsulation: ENET2
  Input packets : 108
  Output packets: 1
  Security: Zone: Null
  Protocol inet, MTU: 8996
    Flags: Sendbcast-pkt-to-re
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 14.1.1/24, Local: 14.1.1.1, Broadcast: 14.1.1.255
```

Verify the following information in the command output:

- Physical interface—ge-2/0/0, Enabled, Physical link is Up
- MTU—9010; Link-mode—Full-duplex
- Speed—100 Mbps
- Loopback—Disabled

Understanding the 2-Port 10-Gigabit Ethernet XPIM

IN THIS SECTION

The 10-Gigabit Ethernet (also known as 10GBASE-T or IEEE 802.3an) is a telecommunication technology that offers data speeds up to 10 billion bits per second over unshielded or shielded twisted pair cables.

The 2-Port 10-Gigabit Ethernet *Physical Interface Module* (XPIM) is a 2 x 10GBASE-T / SFP+ XPIM line card. (SFP+ is a fiber optic transceiver module designed for 10-Gigabit Ethernet and 8.5 Gbps-fiber

channel systems.) The 2-Port 10-Gigabit Ethernet XPIM provides a front-end interface connection that includes the following ports:

- 2 X copper ports. The copper ports support 10GBASE-T running with CAT6A or CAT7 Ethernet cable for up to 100 meters.
- 2 X fiber (SFP+) ports. The fiber ports support SFP+ multiple 10G modules.

The 2-Port 10-Gigabit Ethernet XPIM provides interconnects for LANs, WANs, and metropolitan area networks (MANs). The XPIM provides multiple service levels (1-Gigabit Ethernet to 10-Gigabit Ethernet in increments) and a single connection option for a wide range of customer needs and applications.

NOTE: By default, the 2-Port 10-Gigabit Ethernet XPIM ports comes up in fiber mode, while autonegotiation is not supported.

This topic includes the following sections:

Supported Features

The following features are supported on the 2-Port 10-Gigabit Ethernet XPIM:

- Multiple SFP+ 10G modules and the following SFP modules:
  - SFPP-10GE-SR
  - SFPP-10GE-LR
  - SFPP-10GE-ER
  - SFPP-10GE-LRM
- Copper TWIN-AX 1M and Copper TWIN-AX 3M
- Online Insertion and Removal (OIR) functionality
- Link speeds of up to 10-Gbps
- Full-duplex and half-duplex modes
- Flow control
- Autonegotiation and autosensing
- *Quality of service* (QoS)

Interface Names and Settings

The following format is used to represent the 2-Port 10-Gigabit Ethernet XPIM interface names:

type-fpc/pic/port

Where:

- type — Media type (xe)
- fpc — Number of the Flexible PIC Concentrator (FPC) card on which the physical interface is located
- pic — Number of the PIC on which the physical interface is located (0)
- port — Specific port on a PIC (0 or 1)

By default, the interfaces (for example, `xe-6/0/0` or `xe-2/0/0`) on the ports on the uplink module installed on the device are enabled. You can also specify the maximum transmission unit (MTU) size for the Gigabit Ethernet interface. Junos OS supports values from 256 through 9192. The default MTU for Gigabit Ethernet interfaces is 1514.

Copper and Fiber Operating Modes

On the 2-Port 10-Gigabit Ethernet XPIM, one copper port and one fiber port is grouped together as port 0, and another copper port and fiber port are grouped as port 1. Only two ports can be active at the same time (one port from port 0 and another port from port 1).

The 2-Port 10-Gigabit Ethernet XPIM can be configured to operate in two copper mode, two fiber mode, or mixed mode (one copper and one fiber). In mixed mode, the two ports should be from different port groups (one port from port 1 and the other from port 2).

Link Speeds

The 2-Port 10-Gigabit Ethernet XPIM ports support the following link speeds for copper and fiber:

- Copper—10/100/1000 Mbps or 10Gbps (full duplex). Half-duplex is only for 10/100 Mbps.
- Fiber—1000 Mbps or 10 Gbps (full duplex). Half-duplex mode is not supported.

To set the link speeds, use the following options:

- `10m`—Sets the link speed to 10 Mbps.
- `10g`—Sets the link speed to 10 Gbps.
- `100m`—Sets the link speed to 100 Mbps.

- `1g`—Sets the link speed to 1 Gbps.

Link Settings

The 2-Port 10-Gigabit Ethernet XPIM includes the following link settings:

- `802.3ad`—Specifies an aggregated Ethernet bundle.
- `auto-negotiation`—Enables autonegotiation of flow control, link mode, and speed.
- `loopback`—Enables loopback.
- `no-auto-negotiation`—Disables autonegotiation of flow control, link mode, and speed.
- `no-loopback`—Disables loopback.

By default, flow control is enabled on all ports, a link speed of 10 Gbps in full duplex is supported, autonegotiation is disabled on the fiber ports, and autonegotiation is enabled on copper ports.

NOTE: Autonegotiation is not supported when the 2-Port 10-Gigabit Ethernet XPIM is operating in fiber mode at a link speed of 10 Gbps.

Example: Configuring the 2-Port 10-Gigabit Ethernet XPIM Interface

IN THIS SECTION

This example shows how to perform basic configuration for the 1-Port Gigabit Ethernet SFP Mini-PIM.

Requirements

Before you begin:

- Establish basic connectivity. See the *Getting Started Guide* for your device.

- Configure network interfaces as necessary. See "Example: Creating an Ethernet Interface" on page 130.

Overview

In this example, you configure the xe-6/0/0 interface, set the operating mode to copper mode, set the operating speed to 10 Gbps, and define a logical interface that you can connect to the 2-Port 10-Gigabit Ethernet XPIM. Additionally, you set the MTU value to 1514, set the link option to no loopback, and enable the interface.

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

```
set interfaces xe-6/0/0 media-type copper speed 10g unit 0 family inet mtu 1514
set interface xe-6/0/0 gigether-options no-loopback
```

Step-by-Step Procedure

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

To configure a 2-Port 10-Gigabit Ethernet XPIM:

1. Configure the interface.

```
[edit]
user@host# edit interfaces xe-6/0/0
```

2. Configure the operating mode.

```
[edit interfaces xe-6/0/0]
user@host# set media-type copper
```

3. Set the operating speed for the XPIM.

```
[edit interfaces xe-6/0/0]
user@host# set speed 10g
```

4. Add the logical interface.

```
[edit interfaces xe-6/0/0]
user@host# set unit 0 family inet
```

5. Assign the physical interface MTU value.

```
[edit interfaces xe-6/0/0]
user@host# set interface xe-6/0/0 mtu 1514
```

6. Assign the logical interface MTU value.

```
[edit interfaces xe-6/0/0]
user@host# set unit 0 family inet mtu 1500
```

7. Set the link options.

```
[edit interfaces xe-6/0/0]
user@host# set gigether-options no-loopback
```

8. Disable the interface.

```
[edit interfaces xe-6/0/0]
user@host# set disable
```

9. Enable the interface.

```
[edit interfaces xe-6/0/0]
user@host# delete disable
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces xe-6/0/0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interfaces xe-6/0/0
    speed 10g;
    media-type copper;
    gigether-options {
    no-loopback;
}
unit 0 {
    family inet {
    mtu 1514;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying That the Correct Hardware Is Installed

Purpose

Verify that the 2-Port 10-Gigabit Ethernet XPIM is installed on the device.

Action

From operational mode, enter the `show chassis hardware` command.

```
Hardware inventory:
Item                     Version         Part number     Serial number         Description
Chassis                                                    AJ0309AC0047         SRX650
Midplane             REV 04         710-023875         TV3993
System IO             REV 04         710-023209         TV4035                SRXSME System IO
Routing Engine     REV 01         710-023224         DT5109                RE-SRXSME-SRE6
FPC 0                                                                           FPC
PIC 0                                                                           4x GE Base PIC
FPC 2                                                                           FPC
PIC 0                                                                           2x 10G gPIM
FPC 6                                                                           FPC
PIC 0                                                                           2x 10G gPIM
Power Supply 0     REV 01         740-024283         TA00049WSSSS         PS 645W AC
```

Verify that the output contains the following values:

- `FPC 2 , PIC 0`—2x 10G gPIM
- `FPC 6, PIC 0`—2x 10G gPIM

Verifying the FPC Status

Purpose

Verify the FPC status.

Action

From operational mode, enter the `show chassis fpc` command.

```
            Temp         CPU Utilization     (%)     Memory         Utilization (%)
Slot State         (C)         Total Interrupt             DRAM (MB)     Heap Buffer
0 Online         -------------------- CPU less FPC --------------------
1 Empty
2 Online         -------------------- CPU less FPC --------------------
3 Empty
4 Empty
5 Empty
6 Online         -------------------- CPU less FPC --------------------
7 Empty
8 Empty
```

The output should display FPC status as online.

Verifying the Interface Settings

Purpose

Verify that the interface is configured as expected.

Action

From operational mode, enter the `show interface xe-6/0/0` command.

```
Physical interface: xe-6/0/0, Enabled, Physical link is Up
    Interface index: 144, SNMP ifIndex: 501
    Link-level type: Ethernet, MTU: 1514, Link-mode: Full-duplex, Speed: 10Gbps,
    BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
    Source filtering: Disabled, Flow control: Enabled
    Device flags : Present Running
    6 Copyright © 2010, Juniper Networks, Inc.
    Interface flags: SNMP-Traps Internal: 0x0
    Link flags : None
    CoS queues : 8 supported, 8 maximum usable queues
    Current address: 00:1f:12:e0:80:a8, Hardware address: 00:1f:12:e0:80:a8
    Last flapped : 1970-01-01 00:34:22 PST (07:26:29 ago)
```

```
    Input rate : 0 bps (0 pps)
    Output rate : 0 bps (0 pps)
    Active alarms : None
    Active defects : None

Logical interface xe-6/0/0.0 (Index 72) (SNMP ifIndex 503)
    Flags: SNMP-Traps Encapsulation: ENET2
    Input packets : 25
    Output packets: 25
    Security: Zone: HOST
    Allowed host-inbound traffic : any-service bfd bgp dvmrp igmp ldp msdp nhrp
    ospf pgm pim rip router-discovery rsvp sap vrrp
    Protocol inet, MTU: 1500
    Flags: Sendbcast-pkt-to-re
    Addresses, Flags: Is-Preferred Is-Primary
    Destination: 10.10.10/24, Local: 10.10.10.10, Broadcast: 10.10.10.255
```

Verify the following information in the command output:

- Physical interface—xe-6/0/0, Enabled, Physical link is Up
- MTU—1514
- Link mode—Full duplex
- Speed—10 Gbps
- Loopback—Disabled
- Flow control—Enabled

Understanding the 8-Port Gigabit Ethernet SFP XPIM

IN THIS SECTION

A Gigabit Ethernet *Physical Interface Module* (XPIM) is a network interface card (NIC) that installs in the front slots of the SRX550 Services Gateway to provide physical connections to a LAN or a WAN.

NOTE: Starting in Junos OS Release 15.1X49-D10, the 8-Port Gigabit Ethernet SFP XPIM is not supported on legacy SRX Series systems. In Junos OS Release 15.1X49-D30, support for the 8-Port Gigabit Ethernet SFP XPIM is restored for SRX550 Service Gateway systems.

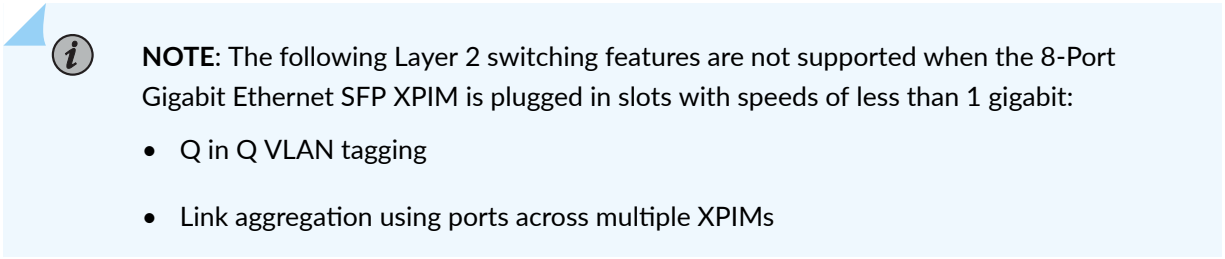
Small form-factor pluggables (SFPs) are hot-pluggable modular interface transceivers for gigabit connections. The 8-port SFP Gigabit Ethernet interface enables customers to connect to Ethernet WAN services as well as to local servers at gigabit speed.

Supported Features

The following features are supported on the 8-Port Gigabit Ethernet SFP XPIM:

- Operates on both a slot with a maximum bandwidth of 8 gigabits and a slot with a maximum bandwidth of 1 gigabit
- Operates in tri-rate (10/100/1000 Mbps) mode with copper SFPs
- Routing and switched mode operation
- Layer 2 protocols
  - Link Aggregation Control Protocol (LACP)
  - Link Layer Discovery Protocol (LLDP)
  - GARP VLAN Registration Protocol (GVRP)
  - Internet Group Management Protocol (IGMP) snooping (v1 and v2)
  - Spanning Tree Protocol (STP), Real-Time Streaming Protocol (RTSP), and Multiple Spanning Tree Protocol (MSTP)
  - 802.1x
- Encapsulation (supported at the Physical Layer)
  - ethernet-bridge
  - ethernet-ccc
  - ethernet-tcc
  - ethernet-vpls

  - extended-vlan-ccc
  - extended-vlan-tcc
  - flexible-ethernet-services
  - vlan-ccc
- Q in Q VLAN tagging
- Integrated routing and bridging (IRB)
- Jumbo frames (9192 byte size)
- *Chassis cluster* switching
- Chassis cluster fabric link using GE ports



NOTE: The following Layer 2 switching features are not supported when the 8-Port Gigabit Ethernet SFP XPIM is plugged in slots with speeds of less than 1 gigabit:

- Q in Q VLAN tagging
- Link aggregation using ports across multiple XPIMs

Interface Names and Settings

The following format is used to represent the 8-Port SFP XPIM:

type-fpc/pic/port

Where:

- type—Media type (ge)
- fpc—Number of the Flexible PIC Concentrator (FPC) card where the physical interface resides
- pic—Number of the PIC where the physical interface resides (0)
- port—Specific port on a PIC (0)

Examples: `ge-1/0/0` and `ge-2/0/0`

By default, the interfaces on the ports on the uplink module installed on the device are enabled. You can also specify the maximum transmission unit (MTU) size for the XPIM. Junos OS supports values from 256 through 9192. The default MTU size for the 8-Port Gigabit Ethernet SFP XPIM is 1514.

Example: Configuring 8-Port Gigabit Ethernet SFP XPIMs

IN THIS SECTION

This example shows how to perform a basic back-to-back device configuration with 8-port Gigabit Ethernet small form-factor pluggable (SFP) XPIMs. It describes a common scenario in which SFP XPIMs are deployed.

NOTE: Starting in Junos OS Release 15.1X49-D10, the 8-Port Gigabit Ethernet SFP XPIM is not supported on legacy SRX Series systems. In Junos OS Release 15.1X49-D30, support for the 8-Port Gigabit Ethernet SFP XPIM is restored for SRX550 Service Gateway systems.

Requirements

This example uses the following hardware and software components:

- Junos OS Release 12.1X44-D10 or later for SRX Series Firewalls.
- Two SRX650 devices connected back-to-back.
- Two 8-port Gigabit Ethernet SFP XPIMs.
- Eight pairs of SFP transceivers as mentioned in 8-Port Gigabit Ethernet SFP XPIM Supported Modules and eight cables to connect them.

Before you begin:

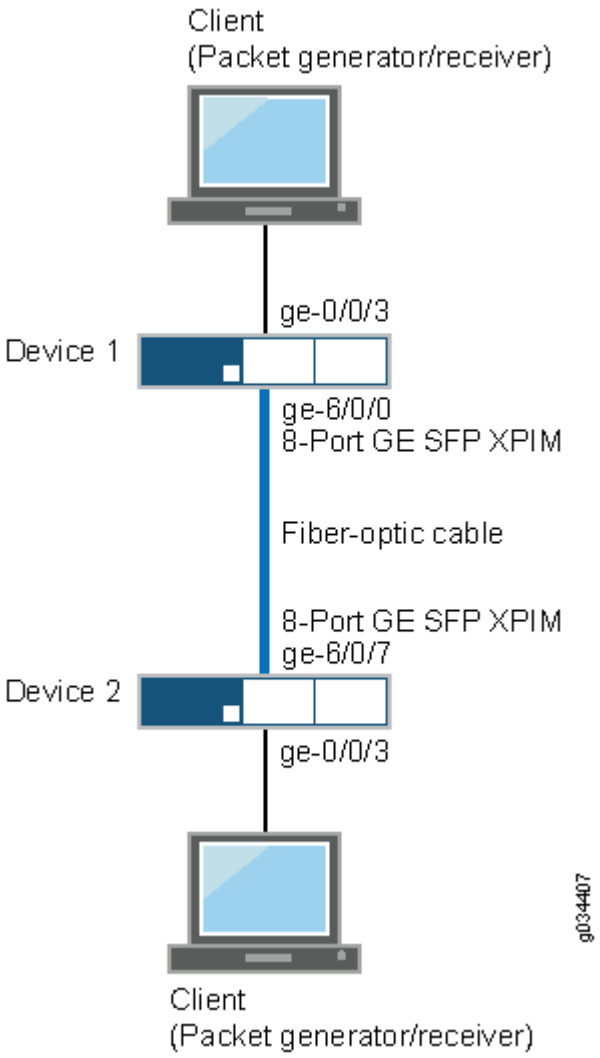
- Establish basic connectivity. See the Getting Started Guide for your device.
- Configure network interfaces as necessary. See "Example: Creating an Ethernet Interface" on page 130.

Overview and Topology

In this example, you configure two SRX650 devices. On each device you configure eight interfaces (ge-6/0/0 through ge-6/0/7), set the maximum transmission unit (MTU) value to 9192, and define a logical interface that you can connect to the 8-port SFP XPIM.

Figure 13 on page 195 shows the topology used in this example.

Figure 13: Basic Back-to-Back Device Configuration



Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

Device 1

```
set interfaces ge-6/0/0 mtu 9192
set interfaces ge-6/0/0 unit 0 family inet address 10.1.1.1/24
set interfaces ge-6/0/1 mtu 9192
set interfaces ge-6/0/1 unit 0 family inet address 11.1.1.1/24
set interfaces ge-6/0/2 mtu 9192
set interfaces ge-6/0/2 unit 0 family inet address 12.1.1.1/24
set interfaces ge-6/0/3 mtu 9192
set interfaces ge-6/0/3 unit 0 family inet address 13.1.1.1/24
set interfaces ge-6/0/4 mtu 9192
set interfaces ge-6/0/4 unit 0 family inet address 14.1.1.1/24
set interfaces ge-6/0/5 mtu 9192
set interfaces ge-6/0/5 unit 0 family inet address 15.1.1.1/24
set interfaces ge-6/0/6 mtu 9192
set interfaces ge-6/0/6 unit 0 family inet address 16.1.1.1/24
set interfaces ge-6/0/7 mtu 9192
set interfaces ge-6/0/7 unit 0 family inet address 17.1.1.1/24
```

Device 2

```
set interfaces ge-6/0/0 mtu 9192
set interfaces ge-6/0/0 unit 0 family inet address 10.1.1.2/24
set interfaces ge-6/0/1 mtu 9192
```

```
set interfaces ge-6/0/1 unit 0 family inet address 11.1.1.2/24
set interfaces ge-6/0/2 mtu 9192
set interfaces ge-6/0/2 unit 0 family inet address 12.1.1.2/24
set interfaces ge-6/0/3 mtu 9192
set interfaces ge-6/0/3 unit 0 family inet address 13.1.1.2/24
set interfaces ge-6/0/4 mtu 9192
set interfaces ge-6/0/4 unit 0 family inet address 14.1.1.2/24
set interfaces ge-6/0/5 mtu 9192
set interfaces ge-6/0/5 unit 0 family inet address 15.1.1.2/24
set interfaces ge-6/0/6 mtu 9192
set interfaces ge-6/0/6 unit 0 family inet address 16.1.1.2/24
set interfaces ge-6/0/7 mtu 9192
set interfaces ge-6/0/7 unit 0 family inet address 17.1.1.2/24
```

Step-by-Step Procedure

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

To configure the interfaces on Device 1:

1. Configure the interface.

```
[edit]
user@host# set interfaces ge-6/0/0
```

2. Assign the maximum transmission unit value for the interface.

```
[edit interfaces ge-6/0/0]
user@host# set mtu 9192
```

3. Add the logical interface.

```
[edit interfaces ge-6/0/0]
user@host# set unit 0 family inet address 10.1.1.1/24
```

NOTE: Repeat these steps for the remaining seven ports on Device 1.

Step-by-Step Procedure

To configure the interfaces on Device 2:

1. Configure the interface.

```
[edit]
user@host# edit interfaces ge-6/0/0
```

2. Assign the maximum transmission unit value for the interface.

```
[edit interfaces ge-6/0/0]
user@host# set mtu 9192
```

3. Add the logical interface.

```
[edit interfaces ge-6/0/0]
user@host# set unit 0 family inet address 10.1.1.2/24
```

NOTE: Repeat these steps for the remaining seven ports on Device 2.

Results

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

Device 1

```
[edit]
user@host# show interfaces
ge-6/0/0 {
    mtu 9192;
    unit 0 {
        family inet {
            address 10.1.1.1/24;
        }
    }
```

```
}
ge-6/0/1 {
    mtu 9192;
    unit 0 {
        family inet {
            address 11.1.1.1/24;
        }
    }
}
ge-6/0/2 {
    mtu 9192;
    unit 0 {
        family inet {
            address 12.1.1.1/24;
        }
    }
}
ge-6/0/3 {
    mtu 9192;
    unit 0 {
        family inet {
            address 13.1.1.1/24;
        }
    }
}
ge-6/0/4 {
    mtu 9192;
    unit 0 {
        family inet {
            address 14.1.1.1/24;
        }
    }
}
ge-6/0/5 {
    mtu 9192;
    unit 0 {
        family inet {
            address 15.1.1.1/24;
        }
    }
}
ge-6/0/6 {
    mtu 9192;
```

```
    unit 0 {
        family inet {
            address 16.1.1.1/24;
        }
    }
}
ge-6/0/7 {
    mtu 9192;
    unit 0 {
        family inet {
            address 17.1.1.1/24;
        }
    }
}
```

Device 2

```
[edit]
user@host# show interfaces
ge-6/0/0 {
    mtu 9192;
    unit 0 {
        family inet {
            address 10.1.1.2/24;
        }
    }
}
ge-6/0/1 {
    mtu 9192;
    unit 0 {
        family inet {
            address 11.1.1.2/24;
        }
    }
}
ge-6/0/2 {
    mtu 9192;
    unit 0 {
        family inet {
            address 12.1.1.2/24;
        }
    }
```

```
}
ge-6/0/3 {
    mtu 9192;
    unit 0 {
        family inet {
            address 13.1.1.2/24;
        }
    }
}
ge-6/0/4 {
    mtu 9192;
    unit 0 {
        family inet {
            address 14.1.1.2/24;
        }
    }
}
ge-6/0/5 {
    mtu 9192;
    unit 0 {
        family inet {
            address 15.1.1.2/24;
        }
    }
}
ge-6/0/6 {
    mtu 9192;
    unit 0 {
        family inet {
            address 16.1.1.2/24;
        }
    }
}
ge-6/0/7 {
    mtu 9192;
    unit 0 {
        family inet {
            address 17.1.1.2/24;
        }
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Hardware was Properly Installed

Purpose

Verify that the 8-Port Gigabit Ethernet SFP XPIM is installed on the device.

Action

From operational mode, enter the `show chassis hardware` command.

```
user@host> show chassis hardware detail
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                AJ3009AA0001      SRX650
Midplane         REV 08   710-023875   AAAK0059
System IO        REV 08   710-023209   AAAJ9290          SRXSME System IO
Routing Engine   REV 13   750-023223   AAAJ1987          RE-SRXSME-SRE6
  ad0    2000 MB  CF 2GB               2009A     0000194075 Compact Flash
  usb0 (addr 1)  DWC OTG root hub 0    vendor 0x0000     uhub0
  usb0 (addr 2)  product 0x005a 90     vendor 0x0409     uhub1
FPC 0                                                    FPC
  PIC 0                                                  4x GE Base PIC
FPC 1            REV 03   750-038290   AADL2016          FPC
FPC 5                                                    FPC
```

```
  PIC 0                                                 8x GE SFP gPIM
FPC 6            REV 03   750-037551   AAEC8065        FPC
  PIC 0                                                 8x GE SFP gPIM
    Xcvr 0       REV 01   740-013111   8043353         SFP-T
    Xcvr 1                NON-JNPR     PC602QW         SFP-SX
    Xcvr 2       k        NON-JNPR     BDS3I           SFP-1000BASE-BX10-D
    Xcvr 3       REV 01   740-011612   9XT702501080    SFP-LH
    Xcvr 4       REV 01   740-011612   9XT702501079    SFP-LH
    Xcvr 5                NON-JNPR     PCH2GTJ         SFP-SX
    Xcvr 6                NON-JNPR     PC604DL         SFP-SX
    Xcvr 7       REV 01   740-011620   5349504         SFP-FX
FPC 8            REV 00   750-038290                   FPC
Power Supply 0
```

Meaning

The output displays the hardware details of the device and a list of all interfaces configured.

Verify that the output contains the following values:

- `FPC 5, PIC 0` —8x SFP gPIM
- `FPC 6, PIC 0` —8x SFP gPIM

NOTE: In the example, the output for 8-Port SFP Gigabit Ethernet XPIM is displayed as 8x GE SFP gPIM.

Verifying the FPC Status

Purpose

Verify that the status of the Flexible PIC Concentrator is online.

Action

From operational mode, enter the `show chassis fpc pic-status` command.

```
user@host>  show chassis fpc pic-status
Slot 0   Online       FPC
```

```
  PIC 0  Online       4x GE Base PIC
Slot 1   Present      FPC
Slot 5   Online       FPC
  PIC 0  Online       8x GE SFP gPIM
Slot 6   Online       FPC
  PIC 0  Online       8x GE SFP gPIM
Slot 8   Present      FPC
```

Meaning

The output shows the FPC status for slot 5 and slot 6 as online. The 8-Port Gigabit Ethernet SFP XPIM is installed in slot 5 and slot 6 of the device.

Verifying Interface Link Status on Device 1

Purpose

Verify that the interface link status is up.

Action

From operational mode, enter the `show interface terse ge-6/0/*` command.

```
user@host> show interface terse ge-6/0/*
```

Output for Device 1

```
Interface               Admin Link Proto    Local                 Remote
ge-6/0/0                up    up
ge-6/0/0.0              up    up   inet     10.1.1.1/24
ge-6/0/1                up    up
ge-6/0/1.0              up    up   inet     11.1.1.1/24
ge-6/0/2                up    up
ge-6/0/2.0              up    up   inet     12.1.1.1/24
ge-6/0/3                up    up
ge-6/0/3.0              up    up   inet     13.1.1.1/24
ge-6/0/4                up    up
ge-6/0/4.0              up    up   inet     14.1.1.1/24
ge-6/0/5                up    up
```

```
ge-6/0/5.0              up    up   inet     15.1.1.1/24
ge-6/0/6                up    up
ge-6/0/6.0              up    up   inet     16.1.1.1/24
ge-6/0/7                up    up
ge-6/0/7.0              up    up   inet     17.1.1.1/24
```

Meaning

The output displays a list of all interfaces configured.

If the link displays `up` for all interfaces, the configuration is working properly. This verifies that the XPIM is up and end-to-end ping is working.

Verifying the Interface Settings on Device 1

Purpose

Verify that the interfaces are configured as expected.

Action

From operational mode, enter the `show interface ge-6/0/0 extensive | no-more` command.

```
user@host>show interface ge-6/0/0 extensive | no-more
```

Output for Device 1

```
Physical interface: ge-6/0/0, Enabled, Physical link is Up
  Interface index: 152, SNMP ifIndex: 544, Generation: 155
  Link-level type: Ethernet, MTU: 9192, Link-mode: Full-duplex, Speed: 1000mbps,
  BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:26:88:04:0a:a8, Hardware address: 00:26:88:04:0a:a8
```

```
Last flapped   : 2012-07-05 21:58:46 PDT (00:13:29 ago)
Statistics last cleared: Never
Traffic statistics:
 Input  bytes  :                 228                   0 bps
 Output bytes  :                 540                   0 bps
 Input  packets:                   3                   0 pps
 Output packets:                   6                   0 pps
Input errors:
  Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0,
  L3 incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0,
  FIFO errors: 0, Resource errors: 0
Output errors:
  Carrier transitions: 1, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0,
  FIFO errors: 0, HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
Egress queues: 8 supported, 4 in use
Queue counters:       Queued packets  Transmitted packets      Dropped packets
  0 best-effort                    3                    3                    0
  1 expedited-fo                   0                    0                    0
  2 assured-forw                   0                    0                    0
  3 network-cont                   0                    0                    0
Queue number:         Mapped forwarding classes
  0                   best-effort
  1                   expedited-forwarding
  2                   assured-forwarding
  3                   network-control
Active alarms  : None
Active defects : None
MAC statistics:                      Receive         Transmit
  Total octets                           268              268
  Total packets                            3                3
  Unicast packets                          3                2
  Broadcast packets                        0                1
  Multicast packets                        0                0
  CRC/Align errors                         0                0
  FIFO errors                              0                0
  MAC control frames                       0                0
  MAC pause frames                         0                0
  Oversized frames                         0
  Jabber frames                            0
  Fragment frames                          0
  VLAN tagged frames                       0
  Code violations                          0
Filter statistics:
```

```
  Input packet count                      0
  Input packet rejects                    0
  Input DA rejects                        0
  Input SA rejects                        0
  Output packet count                                0
  Output packet pad count                            0
  Output packet error count                          0
  CAM destination filters: 2, CAM source filters: 0
Autonegotiation information:
  Negotiation status: Complete
  Link partner:
      Link mode: Full-duplex, Flow control: None, Remote fault: OK,
      Link partner Speed: 1000 Mbps
  Local resolution:
      Flow control: None, Remote fault: Link OK
Packet Forwarding Engine configuration:
  Destination slot: 6
CoS information:
  Direction : Output
  CoS transmit queue              Bandwidth              Buffer Priority   Limit
                            %            bps     %           usec
  0 best-effort            95      950000000    95              0      low    none
  3 network-control         5       50000000     5              0      low    none
Interface transmit statistics: Disabled

Logical interface ge-6/0/0.0 (Index 81) (SNMP ifIndex 509) (Generation 146)
  Flags: SNMP-Traps 0x0 Encapsulation: ENET2
  Traffic statistics:
   Input  bytes  :                   0
   Output bytes  :                  42
   Input  packets:                   0
   Output packets:                   1
  Local statistics:
   Input  bytes  :                   0
   Output bytes  :                  42
   Input  packets:                   0
   Output packets:                   1
  Transit statistics:
   Input  bytes  :                   0                  0 bps
   Output bytes  :                   0                  0 bps
   Input  packets:                   0                  0 pps
   Output packets:                   0                  0 pps
  Security: Zone: HOST
```

```
Allowed host-inbound traffic : any-service bfd bgp dvmrp igmp ldp msdp nhrp
ospf ospf3 pgm pim rip ripng router-discovery rsvp sap vrrp
Flow Statistics :
Flow Input statistics :
  Self packets :                    0
  ICMP packets :                    0
  VPN packets :                     0
  Multicast packets :               0
  Bytes permitted by policy :       0
  Connections established :         0
Flow Output statistics:
  Multicast packets :               0
  Bytes permitted by policy :       0
Flow error statistics (Packets dropped due to):
  Address spoofing:                 0
  Authentication failed:            0
  Incoming NAT errors:              0
  Invalid zone received packet:     0
  Multiple user authentications:    0
  Multiple incoming NAT:            0
  No parent for a gate:             0
  No one interested in self packets: 0
  No minor session:                 0
  No more sessions:                 0
  No NAT gate:                      0
  No route present:                 0
  No SA for incoming SPI:           0
  No tunnel found:                  0
  No session for a gate:            0
  No zone or NULL zone binding      0
  Policy denied:                    0
  Security association not active:  0
  TCP sequence number out of window: 0
  Syn-attack protection:            0
  User authentication errors:       0
Protocol inet, MTU: 9178, Generation: 162, Route table: 0
  Flags: Sendbcast-pkt-to-re
  Addresses, Flags: Is-Preferred Is-Primary
    Destination: 10.1.1/24, Local: 10.1.1.1, Broadcast: 10.1.1.255,
    Generation: 176
```

Meaning

The output displays a list of all interface verification parameters.

Verify the following information in the command output:

- Physical Interface—ge-6/0/0, enabled, physical link is `Up`
- MTU—9192
- Speed—1000 Mbps

If the verification parameters are as expected, the configuration is working properly.

Verifying Interface Link Status on Device 2

Purpose

Verify that the interface link status is up.

Action

From operational mode, enter the `show interface terse ge-6/0/*` command.

```
user@host> show interface terse ge-6/0/*
```

Output for Device 2

```
Interface               Admin Link Proto    Local                 Remote
ge-6/0/0                up    up
ge-6/0/0.0              up    up   inet     10.1.1.2/24
ge-6/0/1                up    up
ge-6/0/1.0              up    up   inet     11.1.1.2/24
ge-6/0/2                up    up
ge-6/0/2.0              up    up   inet     12.1.1.2/24
ge-6/0/3                up    up
ge-6/0/3.0              up    up   inet     13.1.1.2/24
ge-6/0/4                up    up
ge-6/0/4.0              up    up   inet     14.1.1.2/24
ge-6/0/5                up    up
ge-6/0/5.0              up    up   inet     15.1.1.2/24
ge-6/0/6                up    up
```

```
ge-6/0/6.0              up    up   inet     16.1.1.2/24
ge-6/0/7                up    up
ge-6/0/7.0              up    up   inet     17.1.1.2/24
```

Meaning

The output displays a list of all interfaces configured.

If the link displays `up` for all interfaces, the configuration is working properly. This verifies that the XPIM is up and end-to-end ping is working.

Verifying the Interface Settings on Device 2

Purpose

Verify that the interfaces are configured as expected.

Action

From operational mode, enter the `show interface ge-6/0/0 extensive | no-more` command.

```
user@host>show interface ge-6/0/0 extensive | no-more
```

Output for Device 2

```
Physical interface: ge-6/0/0, Enabled, Physical link is Up
  Interface index: 144, SNMP ifIndex: 520, Generation: 147
  Link-level type: Ethernet, MTU: 9192, Link-mode: Full-duplex, Speed: 1000mbps,
  BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:24:dc:17:2f:a8, Hardware address: 00:24:dc:17:2f:a8
  Last flapped   : 2012-07-05 21:59:42 PDT (00:15:32 ago)
  Statistics last cleared: Never
```

```
Traffic statistics:
 Input  bytes  :                 228                   0 bps
 Output bytes  :                 294                   0 bps
 Input  packets:                   3                   0 pps
 Output packets:                   5                   0 pps
Input errors:
  Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0,
  L3 incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0,
  FIFO errors: 0, Resource errors: 0
Output errors:
  Carrier transitions: 13, Errors: 0, Drops: 0, Collisions: 0,
  Aged packets: 0, FIFO errors: 0, HS link CRC errors: 0, MTU errors: 0,
  Resource errors: 0
Egress queues: 8 supported, 4 in use
Queue counters:       Queued packets  Transmitted packets      Dropped packets
  0 best-effort                   3                     3                    0
  1 expedited-fo                  0                     0                    0
  2 assured-forw                  0                     0                    0
  3 network-cont                  0                     0                    0
Queue number:         Mapped forwarding classes
  0                   best-effort
  1                   expedited-forwarding
  2                   assured-forwarding
  3                   network-control
Active alarms  : None
Active defects : None
MAC statistics:                      Receive         Transmit
  Total octets                           268              268
  Total packets                            3                3
  Unicast packets                          2                3
  Broadcast packets                        1                0
  Multicast packets                        0                0
  CRC/Align errors                         0                0
  FIFO errors                              0                0
  MAC control frames                       0                0
  MAC pause frames                         0                0
  Oversized frames                         0
  Jabber frames                            0
  Fragment frames                          0
  VLAN tagged frames                       0
  Code violations                          0
Filter statistics:
  Input packet count                       0
```

```
  Input packet rejects                    0
  Input DA rejects                        0
  Input SA rejects                        0
  Output packet count                                    0
  Output packet pad count                                0
  Output packet error count                              0
  CAM destination filters: 2, CAM source filters: 0
Autonegotiation information:
  Negotiation status: Complete
  Link partner:
      Link mode: Full-duplex, Flow control: None, Remote fault: OK,
      Link partner Speed: 1000 Mbps
  Local resolution:
      Flow control: None, Remote fault: Link OK
Packet Forwarding Engine configuration:
  Destination slot: 6
CoS information:
  Direction : Output
  CoS transmit queue               Bandwidth               Buffer Priority   Limit
                            %            bps     %           usec
  0 best-effort            95      950000000    95              0      low    none
  3 network-control         5       50000000     5              0      low    none
Interface transmit statistics: Disabled

Logical interface ge-6/0/0.0 (Index 73) (SNMP ifIndex 509) (Generation 146)
  Flags: SNMP-Traps 0x0 Encapsulation: ENET2
  Traffic statistics:
   Input  bytes  :                   0
   Output bytes  :                  42
   Input  packets:                   0
   Output packets:                   1
  Local statistics:
   Input  bytes  :                   0
   Output bytes  :                  42
   Input  packets:                   0
   Output packets:                   1
  Transit statistics:
   Input  bytes  :                   0                  0 bps
   Output bytes  :                   0                  0 bps
   Input  packets:                   0                  0 pps
   Output packets:                   0                  0 pps
  Security: Zone: HOST
  Allowed host-inbound traffic : any-service bfd bgp dvmrp igmp ldp msdp nhrp
```

```
ospf ospf3 pgm pim rip ripng router-discovery rsvp sap vrrp
Flow Statistics :
Flow Input statistics :
  Self packets :                    0
  ICMP packets :                    0
  VPN packets :                     0
  Multicast packets :               0
  Bytes permitted by policy :       0
  Connections established :         0
Flow Output statistics:
  Multicast packets :               0
  Bytes permitted by policy :       0
Flow error statistics (Packets dropped due to):
  Address spoofing:                 0
  Authentication failed:            0
  Incoming NAT errors:              0
  Invalid zone received packet:     0
  Multiple user authentications:    0
  Multiple incoming NAT:            0
  No parent for a gate:             0
  No one interested in self packets: 0
  No minor session:                 0
  No more sessions:                 0
  No NAT gate:                      0
  No route present:                 0
  No SA for incoming SPI:           0
  No tunnel found:                  0
  No session for a gate:            0
  No zone or NULL zone binding      0
  Policy denied:                    0
  Security association not active:  0
  TCP sequence number out of window: 0
  Syn-attack protection:            0
  User authentication errors:       0
Protocol inet, MTU: 9178, Generation: 162, Route table: 0
  Flags: Sendbcast-pkt-to-re
  Addresses, Flags: Is-Preferred Is-Primary
    Destination: 10.1.1/24, Local: 10.1.1.2, Broadcast: 10.1.1.255,
    Generation: 176
```

Meaning

The output displays a list of all interface verification parameters.

Verify the following information in the command output:

- Physical Interface—ge-6/0/0, enabled, physical link is `Up`
- MTU—9192
- Speed—1000 Mbps

If the verification parameters are as expected, the configuration is working properly.

Change History Table

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

Release	Description
15.1X49-D10	Starting in Junos OS Release 15.1X49-D10, the 8-Port Gigabit Ethernet SFP XPIM is not supported on legacy SRX Series systems.
15.1X49-D10	Starting in Junos OS Release 15.1X49-D10, the 8-Port Gigabit Ethernet SFP XPIM is not supported on legacy SRX Series systems.

RELATED DOCUMENTATION

Ethernet Interfaces Overview | 125

Port Speed on SRX Series Firewalls

SUMMARY

Learn about port speeds, support for multiple port speeds, and how to configure port speed on SRX Series Firewalls.

IN THIS SECTION

Port Speed on SRX380 Firewalls

Table 25 on page 215 presents the details of SRX380 speeds.

Table 25: Port Speed Details and Description

Port Location	Number and Type of Ports	Supported Speeds
FPC0, PIC0	16 RJ45, 4 SFP ports	1 Gbps
FPC0, PIC0	4 SFP ports	10 Gbps

Follow the guideline given below when you configure the speed of a port:

- Use the configuration to set 1-Gbps speed on PIC 1 ports. 1-Gbps speed is supported only in non-autonegotiation mode. If autonegotiation mode is enabled by default at the remote end, then you must disable it.

Port Speed on SRX1600 Firewalls

IN THIS SECTION

To view the supported transceivers, optical interfaces, and DAC cables on SRX1600, see Hardware Compatibility Tool (HCT).

SRX1600 includes three PICs with different supported speeds:

- PIC 0 with 1 GbE default speed
- PIC 1 with 25 GbE default speed
- PIC 2 with 10 GbE default speed

See Table 26 on page 216 for details.

Table 26: Port Speed Details and Description for SRX1600

PIC	Port	Port Speed Supported	Default Speed
PIC 0	0-15	16x1-GbE interface	1 Gbps
PIC 1	0-1	2x1-GbE Interface 2x10-GbE Interface 2x25-GbE Interface	25 Gbps
PIC 2	0-3	4x1-GbE Interface 4x10-GbE Interface	10 Gbps

Interface Naming Conventions

Table 27 on page 216 lists the interface naming conventions for the SRX1600 devices.

Table 27: Interface Naming Conventions for SRX1600

PIC	Interface Type	Interfaces
PIC 0	1-Gigabit Ethernet interface (16 RJ45 ports)	ge-0/0/0 – ge-0/0/15
PIC 1	1 GbE/10 GbE/25 GbE (2 SFP28 ports)	et-0/1/0 – et-0/1/1

Table 27: Interface Naming Conventions for SRX1600 *(Continued)*

PIC	Interface Type	Interfaces
PIC 2	1 GbE/10 GbE (4 SFP+ ports)	xe-0/2/0 – xe-0/2/3

Follow these guidelines when you configure the speed of the port:

- To view the port speeds on each PIC, execute the `show chassis pic` command.
- The Junos OS creates PIC 0 interfaces by default. It creates PIC1 and PIC2 interfaces only if you use supported transceivers.
- PIC 1 supports 3 different speed modes: 1 GbE, 10 GbE, and 25 GbE.

  Use the following command to configure the port speed:

  ```
  set chassis fpc 0 pic 1 pic-mode
  ```

  ```
  root@host# set chassis fpc 0 pic 1 pic-mode ?
  Possible completions:
    1G                    1GE mode
    10G                  10GE mode
    25G                  25GE mode
  [edit]
  root@host#
  ```

  You can select any one from the three PIC modes and all the SFP ports run on the speed corresponding to the selected mode. For example, if you select 1GbE mode, the two SFP28 ports run 1GbE speed.

  Starting in Junos OS Release 24.2R1, you have two configuration options in PIC mode: 1GbE/10GbE combined and 25GbE. You can configure the SFP28 ports in two ways:

  - If you select 1GbE/10GbE mode, the SFP28 ports can choose either 1GbE or 10GbE.
  - If you select 25GbE mode, the SFP28 ports run on 25GbE.

  ```
  root@host# set chassis fpc 0 pic 1 pic-mode ?
  Possible completions:
    1G10G                1/10GE mode
    25G                  25GE mode
  ```

```
[edit]
root@host#
```

- PIC 2 supports mixed speed, 1GbE or 10GbE. PIC 2 creates the interface that is based on the plugged-in transceiver.

  See Port Speed Overview to configure speed step-by-step.

- Use the `show interface diagnostics optics <interface-name>` command to display diagnostic data and alarms.

NOTE: If you insert:

- 1 GbE SFP, then the devices show the interfaces as `ge`.
- 10 GbE SFP, then the devices show the interfaces as `xe`.

Port Speed on SRX2300 Firewalls

IN THIS SECTION

To view the supported transceivers, optical interfaces, and DAC cables on SRX2300, see Hardware Compatibility Tool.

SRX2300 includes four PICs with the following properties:

- PIC 0 with default speed of 10 Gbps
- PIC 1 default speed of 10 Gbps
- PIC 2 with default speed of 25 Gbps
- PIC 3 with default speed of 100 Gbps

Table 28: Port Speed Details and Description

Port Location	Number and Type of Ports	Supported Speeds	Default Speed
PIC 0 (ports 0–7)	8 RJ-45 ports	1 Gbps, 2.5 Gbps, 5 Gbps, and 10 Gbps	10 Gbps
PIC 1 (ports 0–7)	8 SFP+ ports	1 Gbps, 10 Gbps	10 Gbps
PIC 2 (ports 0–3)	4 SFP28 ports	1 Gbps, 10 Gbps, and 25 Gbps	25 Gbps
PIC 3 (ports 0–1)	2 QSFP28 ports	40 Gbps, 100 Gbps	100 Gbps

Table 29: Interface Naming Conventions

PIC	Interface Type	Interfaces
PIC 0	RJ-45	mge-0/0/0 – mge-0/0/7
PIC 1	SFP+	xe-0/1/0 – xe-0/1/7
PIC 2	SFP28	et-0/2/0 – et-0/2/3
PIC 3	QSFP28	et-0/3/0 – et-0/3/1

Follow these guidelines when you configure the port speed:

- The Junos OS creates the copper interfaces of PIC 0 (mge interfaces) by default.
- Do not change the speed when using mge interfaces as fabric interfaces to form chassis clusters.
- PIC 0 supports 1 Gbps, 2.5 Gbps, 5 Gbps, and 10 Gbps speeds.

  Use the following command to configure the speed:

  ```
  set interfaces <mge-x/y/z> speed
  ```

- PIC 1 supports mixed speed: 1 Gbps and 10 Gbps. The interface is created based on the plugged-in transceiver. You need not configure the speed.

```
user@host# set chassis fpc 0 pic 1 pic-mode ?
No valid completions
```

- PIC 2 supports 3 different speeds: 1GbE, 10GbE, and 25GbE. Starting in Junos OS Release 24.2R1, you have two configuration options in PIC mode: 1GbE/10GbE combined and 25GbE. You can configure the SFP28 ports in two ways:
  - If you select 1GbE/10GbE mode, the SFP28 ports can choose either 1GbE or 10GbE.
  - If you select 25GbE mode, the SFP28 ports run on 25GbE.

```
user@host# set chassis fpc 0 pic 2 pic-mode ?
Possible completions:
  1G10G                1/10GE mode
  25G                  25GE mode
```

- See Port Speed Overview to configure speed step-by-step.
- PIC 3 supports mixed speed, 40 Gbps or 100 Gbps. The interface is created based on the plugged-in transceiver. You need not configure the speed.

```
user@host# set chassis fpc 0 pic 3 pic-mode ?
No valid completions
```

- Use the `show interface diagnostics optics <interface-name>` command to display diagnostic data and alarms.

Channelization

You can channelize QSFP28 ports into:

- 4x25 Gbps with 100GbE SFP
- 4x10 Gbps with 40GbE SFP

`set chassis fpc <fpc slot> pic <pic slot> port <port number> channel-speed <10G | 25G>`

Example:

```
set chassis fpc 0 pic 3 port 4 channel-speed 25G

set chassis fpc 0 pic 3 port 4 channel-speed 10G
```

NOTE: If you insert:

- 1 GbE SFP, then the devices show the interfaces as `ge`.
- 10 GbE SFP, then the devices show the interfaces as `xe`.

Port Speed on SRX4120 Firewalls

SUMMARY

Learn about supported speeds, default speed, and interface naming conventions of SRX4210 firewalls.

IN THIS SECTION

To view the supported transceivers, optical interfaces, and DAC cables on SRX4120, see Hardware Compatibility Tool.

SRX2300 includes four PICs with the following properties:

- PIC 0 with default speed of 10 Gbps
- PIC 1 default speed of 10 Gbps
- PIC 2 with default speed of 25 Gbps
- PIC 3 with default speed of 100 Gbps

Table 30: Port Speed Details and Description

Port Location	Number and Type of Ports	Supported Speeds	Default Speed
PIC 0 (ports 0–7)	8 RJ-45 ports	1 Gbps, 2.5 Gbps, 5 Gbps, and 10 Gbps	10 Gbps
PIC 1 (ports 0–7)	8 SFP+ ports	1 Gbps, 10 Gbps	10 Gbps

Table 30: Port Speed Details and Description *(Continued)*

Port Location	Number and Type of Ports	Supported Speeds	Default Speed
PIC 2 (ports 0–3)	4 SFP28 ports	1 Gbps, 10 Gbps, and 25 Gbps	25 Gbps
PIC 3 (ports 0–1)	2 QSFP28 ports	40 Gbps, 100 Gbps	100 Gbps

Table 31: Interface Naming Conventions

PIC	Interface Type	Interfaces
PIC 0	RJ-45	mge-0/0/0 – mge-0/0/7
PIC 1	SFP+	xe-0/1/0 – xe-0/1/7
PIC 2	SFP28	et-0/2/0 – et-0/2/3
PIC 3	QSFP28	et-0/3/0 – et-0/3/1

Follow these guidelines when you configure the port speed:

- The Junos OS creates the copper interfaces of PIC 0 (mge interfaces) by default.
- Do not change the speed when using mge interfaces as fabric interfaces to form chassis clusters.
- PIC 0 supports 1 Gbps, 2.5 Gbps, 5 Gbps, and 10 Gbps speeds.

  Use the following command to configure the speed:

  ```
  set interfaces <mge-x/y/z> speed
  ```

- PIC 1 supports mixed speed: 1 Gbps and 10 Gbps. The interface is created based on the plugged-in transceiver. You need not configure the speed.

```
user@host# set chassis fpc 0 pic 1 pic-mode ?
No valid completions
```

- PIC 2 supports 3 different speeds: 1 Gbps, 10 Gbps, and 25 Gbps. Starting in Junos OS Release 24.2R1, you have two configuration options in PIC mode: 1 Gbps/10 Gbps combined and 25 Gbps. You can configure the SFP28 ports in two ways:
  - If you select 1 Gbps/10 Gbps mode, the SFP28 ports can choose either 1 Gbps or 10 Gbps.
  - If you select 25 Gbps mode, the SFP28 ports run on 25 Gbps.

```
user@host# set chassis fpc 0 pic 2 pic-mode ?
Possible completions:
  1G10G                1/10GE mode
  25G                  25GE mode
```

- PIC 3 supports mixed speed, 40 Gbps or 100 Gbps. The interface is created based on the plugged-in transceiver. You need not configure the speed.

```
user@host# set chassis fpc 0 pic 3 pic-mode ?
No valid completions
```

- Use the `show interface diagnostics optics <interface-name>` command to display diagnostic data and alarms.

Channelization

You can channelize QSFP28 ports into:

- 4x25 Gbps with 100GbE SFP
- 4x10 Gbps with 40GbE SFP

```
set chassis fpc <fpc slot> pic <pic slot> port <port number> channel-speed <10G | 25G>
```

Example:

```
set chassis fpc 0 pic 3 port 4 channel-speed 25G

set chassis fpc 0 pic 3 port 4 channel-speed 10G
```

NOTE: If you insert:

- 1 GbE SFP, then the devices show the interfaces as `ge`.

- 10 GbE SFP, then the devices show the interfaces as xe.

Port Speed on SRX4300 Firewalls

IN THIS SECTION

To view the supported transceivers, optical interfaces, and DAC cables on SRX4300, see Hardware Compatibility Tool.

SRX4300 includes four PICs with default speeds as given below:

- PIC 0 with 10 GbE
- PIC 1 with 10 GbE
- PIC 2 with 25 GbE
- PIC 3 with 100 GbE

Table 32: Port Speed Details and Description for SRX4300

Port Location	Number and Type of Ports	Supported Speeds	Default Speed
PIC 0	Eight RJ45 ports	1 GbE, 2.5 GbE, 5 GbE, and 10 GbE	10 GbE
PIC 1	Eight SFP+ ports	1 GbE and 10 GbE	10 GbE
PIC 2	Four SFP28 ports	1 GbE, 10 GbE, and 25 GbE	25 GbE
PIC 3	Six QSFP28 ports	40 GbE and 100 GbE	100 GbE

Table 33: Interface Naming Conventions for SRX4300

PIC	Interface Type	Naming Format
PIC 0	RJ45	mge-0/0/0 - mge-0/0/7
PIC 1	SFP+	xe-0/1/0 - xe-0/1/7
PIC 2	SFP28	et-0/2/0 - et-0/2/3
PIC 3	QSFP28	et-0/3/0 - et-0/3/5

Follow the guidelines below to configure the port speed:

- The Junos OS creates the copper interfaces of PIC 0 (mge interfaces) by default.
- You can create the PIC 1, PIC 2, and PIC 3 (SFP+, SFP28, and QSFP28) interfaces based on the plugged-in transceivers.
- You need not configure the speed for PIC 1. For example:

```
root@srx4300# set chassis fpc 0 pic 1 pic-mode ?
No valid completions
```

- PIC 2 supports three different speed modes: 1 GbE, 10 GbE, and 25 GbE.

  Configure the speed:

```
root@srx4300# set chassis fpc 0 pic 2 pic-mode ?
Possible completions:
  10g                  Sets the interface mode to 10Gbps
  1g                   1GE-Gigabit Ethernet
  25g                  Sets the interface mode to 25Gbps
```

  You can select any one from the three PIC modes and all the SFP ports run on the speed corresponding to the selected mode. For example, if you select 1GbE mode, the four SFP28 ports run 1GbE speed.

  Starting in Junos OS Release 24.2R1, you have two configuration options in PIC mode: 1GbE/10GbE combined and 25GbE. You can configure the SFP28 ports in two ways:

- If you select 1GbE/10GbE mode, the SFP28 ports can choose either 1GbE or 10GbE.
- If you select 25GbE mode, all the SFP28 ports run on 25GbE.

```
root@host# set chassis fpc 0 pic 2 pic-mode ?
Possible completions:
  1G10G                1/10GE mode
  25G                  25GE mode
[edit]
root@host#
```

- PIC 3 supports mixed speed, 40 Gbps or 100 Gbps. The interface is created based on the plugged-in transceiver. You need not configure the speed.

```
user@host# set chassis fpc 0 pic 3 pic-mode ?
No valid completions
```

- Use the `show interface diagnostics optics <interface-name>` command to display diagnostic data and alarms.
- 

  NOTE: If you insert:
  - 1 GbE SFP, then the devices show the interfaces as `ge`.
  - 10 GbE SFP, then the devices show the interfaces as `xe`.

Channelization

You can channelize QSFP28 ports into:

- 4x25 Gbps with 100GbE SFP
- 4x10 Gbps with 40GbE SFP

```
set chassis fpc <fpc slot> pic <pic slot> port <port number> channel-speed <10G | 25G>
```

Example:

```
set chassis fpc 0 pic 3 port 4 channel-speed 25G
```

```
set chassis fpc 0 pic 3 port 4 channel-speed 10G
```

Port Speed on SRX4600 Firewalls

IN THIS SECTION

Table 34 on page 227 presents the details of SRX4600 port speeds.

For information about interface-naming formats for channelized and nonchannelized interfaces and how to configure SRX Series Firewalls at port level and PIC level, see *Port Speed*.

For information on how to configure the speed at the PIC level, see Table 2 of port speed. For information on how to configure the speed at the port level, see Table 3 of port speed.

For more information about SRX4600 devices, see SRX4600 Services Gateway Hardware Guide.

For information about platforms support, see hardware compatibility tool (HCT).

To view the port speeds on each PIC, execute the show chassis pic command.

Table 34: Port Speed Details and Description

Port Location	Number and Type of Ports	Supported Speeds
FPC0, PIC0 (ports **0–3**)	4 chassis cluster ports: • 2 fabric (FAB) • 2 control (CTL)	• 10 Gbps (default) • 1 Gbps (only on CTL ports)
FPC1, PIC0 (ports **0–3**)	4 100GbE QSFP28 ports or 40GbE QSFP+ ports	At port or PIC level: • 40 Gbps (default), with QSFP+ optics • 100 Gbps, with QSFP28 optics

Table 34: Port Speed Details and Description *(Continued)*

Port Location	Number and Type of Ports	Supported Speeds
FPC1, PIC1 (ports 0–7)	8 10GbE SFP+ ports	• 10 Gbps (default) • 1 Gbps

Follow these guidelines when you configure the speed of a port:

- You need to reboot the chassis cluster for configuration changes (from 10 Gbps to 1 Gbps) to take effect. For more details, see speed (Chassis Cluster).
- To configure all 40GbE ports, use the `set chassis fpc 1 pic 0 pic-mode 40G` command.
- To set only the first two 40GbE ports, use the `set chassis fpc 1 pic 0 pic-mode 40G number-of-ports 2` command. This configuration sets only the first two 40GbE ports and disables the last two ports. You need to reboot the device for the configuration to take effect.
- You can channelize each 40GbE port into four 10GbE interfaces by using QSFPP-4X10-GE optics, suitable breakout cables, and the speed configuration statement.
- Use the speed (Gigabit Ethernet interface) configuration to set 1-Gbps speed on PIC 1 ports. 1-Gbps speed is supported only in non-autonegotiation mode. If autonegotiation mode is enabled by default at the remote end, then you must disable it.
- You can configure the interface that is already operating in 10GbE mode to operate in 1GbE mode.
- To prevent oversubscription, configure the number of active ports operating at the configured speed by using the number-of-ports statement. The SRX4600 supports a maximum speed of 400 Gbps; the speed cannot be oversubscribed.
- To configure 4x100GbE, use the following commands:

```
set chassis fpc 1 pic 0 port 0 speed 100g

set chassis fpc 1 pic 0 port 1 speed 100g

set chassis fpc 1 pic 0 port 2 speed 100g

set chassis fpc 1 pic 0 port 3 speed 100g

set chassis fpc 1 pic 1 number-of-ports 0
```

or

```
set chassis fpc 1 pic 0 pic-mode 100G
```

```
set chassis fpc 1 pic 1 number-of-ports 0
```

- If you try to commit an invalid configuration, the configuration gets committed, but the port is not activated. This is because Junos OS allows you to configure a port before a line card is inserted. You will get an error message in the output of the show chassis alarms command and also in the log messages. For example, if you configure four 100GbE interfaces with eight 10GbE interface, then the configuration is invalid.
- SRX4600 supports HA cluster. You need to reboot the system after changing port speed from 40G to 100G. The reboot is to make sure that the system returns to a stable HA cluster after the port speed change.
- The SRX4600 does not support copper SFP transceivers.

Interface Naming Conventions

Table 35 on page 229 describes the interface naming convention for a 40GbE interface channelized as four 10GbE interfaces:

Table 35: SRX4600 Interface Naming Convention

Interface Type	Example
4x10GbE	When the 40GbE port et-1/0/0 is channelized into four 10GbE interfaces, the channelized interfaces are named as follows:: xe-1/0/0:0 xe-1/0/0:1 xe-1/0/0:2 xe-1/0/0:3

Supported Active Physical Ports on SRX4600 to Prevent Oversubscription

You can use the number-of-ports statement to configure a port as an active port.

Table 36 on page 230 summarizes the SRX4600 active ports with number-of-ports and port speed configured at PIC level.

Table 36: SRX4600 Port Speed at PIC level

PIC	Number of Active Ports	Active Port Number at PIC Level		
		10-Gigabit Ethernet	40-Gigabit Ethernet	100-Gigabit Ethernet
PIC 0	0	-	-	-
	1	0	0	0
	2	0, 1	0, 1	0, 1
	3	0, 1, 2	0, 1, 2	0, 1, 2
	4	0, 1, 2, 3	0, 1, 2, 3	0, 1, 2, 3
PIC 1	0	-	-	-
	1	0	-	-
	2	0, 1	-	-
	3	0, 1, 2	-	-
	4	0, 1, 2, 3	-	-
	5	0, 1, 2, 3, 4	-	-
	6	0, 1, 2, 3, 4, 5	-	-
	7	0, 1, 2, 3, 4, 5 ,6	-	-
	8	0, 1, 2, 3, 4, 5, 6, 7	-	-

To prevent oversubscription, you can configure the maximum number of active ports that can operate at the configured speed. Table 37 on page 231 summarizes the maximum number of Gigabit Ethernet ports at PIC and port levels:

Table 37: Maximum Number of Gigabit Ethernet Ports at PIC and Port Level

Port Type	Maximum Number of Ports at PIC Mode (on PIC0 and PIC1)	Maximum Ports Configurable at Port Mode (on PIC0 and PIC1)
10GbE	24 16 ports from PIC 0 and 8 ports from PIC 1.	20 Refers to 12 ports from PIC 0 and 8 ports from PIC 1.
40GbE	4 Only 4 ports from PIC 0. PIC 1 supports only 10-Gbps speed.	4
100GbE	4 Only 4 ports from PIC 0. PIC 1 supports only 10-Gbps speed. **NOTE**: If you configure all four PIC 0 ports as 100GbE interfaces then, the PIC 1 ports are disabled. If you then try to configure any PIC 1 port and commit your configuration, the configuration will be invalid.	4

For information about oversubscription, see Port Speed.

Port Speed on SRX4700 Firewalls

SUMMARY

Provides port speeds that are supported on PICs, default port speed, and interface naming convention of SRX4700 series firewalls.

IN THIS SECTION

- 4x10G Channelization on SRX4700 Firewall | 0

To view the supported transceivers, optical interfaces, and DAC cables on SRX4700 firewalls, see Hardware Compatibility Matrix.

SRX4700 Series Firewalls have two identical logical PICs with 14 front panel ports consisting of 1x400G, 5x100G, and 8x50G ports.

Table 38: Port Speed Details and Description

PIC	Port Number	Port Speeds Supported	Default Speed
PIC 0	0 (QSFP56-DD)	400 Gbps	The default speed of port 0 on both PICs depends on the PIC mode. For example: If you select PIC mode A, the default speed is 400 Gbps. If you select PIC mode C, the default speed is 100 Gbps. See Configuration of PIC Mode for more details.
		100 Gbps	
	1-5 (QSFP28)	100 Gbps	100 Gbps
	6-13 (SFP56)	50 Gbps	50 Gbps
PIC 1	0 (QSFP56-DD)	400 Gbps	The default speed of port 0 on both PICs depends on the PIC mode. For example: If you select PIC mode A, the default speed is 400 Gbps. If you select PIC mode C, the default speed is 100 Gbps. See Configuration of PIC Mode for more details.
		100 Gbps	

Table 38: Port Speed Details and Description *(Continued)*

PIC	Port Number	Port Speeds Supported	Default Speed
	1-5 (QSFP28)	100 Gbps	100 Gbps
	6-13 (SFP56)	50 Gbps	50 Gbps

Table 39: Port Speed Support on Each Port in a PIC

Speed/ Port	0	1	2	3	4	5	6	7	8	9	10	11	12	13
400 G	Y													
100 G	Y	Y	Y	Y	Y	Y								
40 G	Y													
50 G							Y	Y	Y	Y	Y	Y	Y	Y
25 G							Y	Y	Y	Y	Y	Y	Y	Y
10 G							Y	Y	Y	Y	Y	Y	Y	Y
1 G							Y	Y	Y	Y	Y	Y	Y	Y

Table 40: Interface Naming Conventions

PIC	Transceiver Type	Speeds Supported	Example
PIC 0/PIC 1	QSFP56-DD	1x400G	`et-1/0/0` or `et-1/1/0`
		1x100G	

Table 40: Interface Naming Conventions *(Continued)*

PIC	Transceiver Type	Speeds Supported	Example
	QSFP28	1x100G	et-1/0/1 to et-1/0/5 or et-1/1/1 to et-1/1/5
	SFP56	1x50G	et-1/0/6 to et-1/0/13 or et-1/1/6 to et-1/1/13

Configuration of PIC Mode

The PIC mode allows ports to be configured with default speeds. The PIC mode allows the following five port speed combinations:

- A-1x400G-1x100G-4x50G
- B-1x400G-2x100G-2x50G
- C-6x100G-2x50G
- D-3x100G-8x50G
- E-4x100G-6x50G

When the SRX4700 system does not have any PIC mode configuration, the system creates ports and port speeds based on the PIC mode configuration C (6x100G-2x50G).

Use the following command to configure the PIC mode:

```
set chassis fpc <fpc-slot> pic <pic-slot > pic-mode <possible-mode>
```

Example:

```
[edit]
user@host# set chassis fpc 1 pic 0 pic-mode ?
Possible completions:
  A-1X400G-1X100G-4X50G  Port0(400G) Port2(100G) Port6-9(50G)
  B-1X400G-2X100G-2X50G  Port0(400G) Port2,4(100G) Port6,8(50G)
  C-6X100G-2X50G         Port0-5(100G) Port10,12(50G)
```

```
  D-3X100G-8X50G          Port0-2(100G) Port6-13(50G)
  E-4X100G-6X50G          Port0-3(100G) Port7,9-13(50G)
[edit]
```

For 24.4R1-S2, reboot the device using the `request vmhost reboot` command as soon as you configure, delete, or change PIC mode on a PIC.

Configuration of Port Profile Mode

Port profile mode allows you to activate a set of ports based on predefined profiles (A to E). Each profile includes default port speeds. Port profile allows ports to be configured with all the supported speeds other than the default speed. You can adjust the speed of individual ports after activation, enhancing the flexibility compared with the PIC mode.

The following are the pre-defined port profiles:

- **A**: 1x400G (QSFP56-DD), 1x100G (QSFP28), 4x50G (SFP56)
- **B**: 1x400G (QSFP56-DD), 2x100G (QSFP28), 2x50G (SFP56)
- **C**: 6x100G (QSFP28), 2x50G (SFP56)
- **D**: 3x100G (QSFP28), 8x50G (SFP56)
- **E**: 4x100G (QSFP28), 6x50G (SFP56)

Table shows the configurable port speeds based on port profiles A-E.

Table 41: Configurable PIC Port Speeds based on Port Profiles

Port Number	Profile A	Profile B	Profile C	Profile D	Profile E
Default Speed	Configurable Speed	Configurable Speed	Configurable Speed	Configurable Speed	Configurable Speed
0 (400 G)	(400 G/100 G/ 40G)	(400 G/100 G/ 40G)	(100G)	(100G)	(100G)
1 (100 G)			(100G)	(100G)	(100G)
2 (100 G)	(100G)	(100G)	(100G)	(100G)	(100G)
3 (100 G)			(100G)		(100G)

Table 41: Configurable PIC Port Speeds based on Port Profiles *(Continued)*

Port Number	Profile A	Profile B	Profile C	Profile D	Profile E
Default Speed	Configurable Speed	Configurable Speed	Configurable Speed	Configurable Speed	Configurable Speed
4 (100 G)		(100G)	(100G)		
5 (100 G)			(100G)		
6 (50 G)	(50 G/25G/ 10 G/1G)	(50 G/25 G/ 10 G/1G)		(50 G/25 G/ 10 G/1G)	
7 (50 G)	(50 G/25 G/ 10 G/1G)			(50 G/25 G/ 10 G/1G)	(50 G/25 G/ 10 G/1G)
8 (50 G)	(50 G/25 G/ 10 G/1G)	(50 G/25 G/ 10 G/1G)		(50 G/25 G/ 10 G/1G)	
9 (50 G)	(50 G/25 G/ 10 G/1G)			(50 G/25 G/ 10 G/1G)	(50 G/25 G/ 10 G/1G)
10 (50 G)			(50 G/25 G/ 10 G/1G)	(50 G/25 G/ 10 G/1G)	(50 G/25 G/ 10 G/1G)
11 (50 G)				(50 G/25 G/ 10 G/1G)	(50 G/25G/ 10 G/1G)
12 (50 G)			(50 G/25 G/ 10 G/1G)	(50 G/25 G/ 10 G/1G)	(50 G/25 G/ 10 G/1G)
13 (50 G)				(50 G/25 G/ 10 G/1G)	(50 G/25 G/ 10 G/1G)

Example

- Configure the port profile from A to E:

```
set chassis fpc < fpc-slot > pic < pic-slot > port-profile < A-E profile >
```

```
user@host# set chassis fpc 1 pic 0 port-profile ?
Possible completions:
A-1X400G-1X100G-4X50G  Port0(400G/100G/40G) Port2(100G) Port6-9(50G/25G/10G/1G)
```

```
B-1X400G-2X100G-2X50G  Port0(400G/100G/40G) Port2,4(100G) Port6,8(50G/25G/10G/1G)
C-6X100G-2X50G       Port0-5(100G) Port10,12(50G/25G/10G/1G)
D-3X100G-8X50G       Port0-2(100G) Port6-13(50G/25G/10G/1G)
E-4X100G-6X50G       Port0-3(100G) Port7,9-13(50G/25G/10G/1G)
```

```
user@host# set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G
```

```
user@host# show chassis
fpc 1 {
    pic 0 {
        port-profile A-1X400G-1X100G-4X50G;
    }
}
```

- Configure port profile and specific port speed:

  ```
  set chassis fpc < fpc-slot > pic < pic-slot > port-profile < A-E profile > port-num <port-number> speed <port-speed>
  ```

```
user@host# set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num ?
Possible completions:
<slot>  Refer 'show chassis pic port-profile' for valid port. Commit port-profile change
first (0..13)
user@host# run show chassis pic port-profile fpc-slot 1 pic-slot 0
Port profile speed information:
Current port profile: A-1X400G-1X100G-4X50G
Port    Current Speed     Default Speed      Configurable Speed
0       400GE                400GE                      40GE, 100GE, 400GE
2       100GE                100GE                      100GE
6       50GE                 50GE                       1GE, 10GE, 25GE, 50GE*
7       50GE                 50GE                       1GE, 10GE, 25GE, 50GE*
8       50GE                 50GE                       1GE, 10GE, 25GE, 50GE*
9       50GE                 50GE                       1GE, 10GE, 25GE, 50GE*
```

  Paired ports (6,8) and (7,9) must be configured to the same speed.

```
user@host# set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 6 speed 10g

user@host# set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 8 speed 10g
```

- Delete speed of port

```
delete chassis fpc < fpc-slot > pic < pic-slot > port-profile < A-E profile > port-num <port-number>
```

```
user@host# show chassis | display set
set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 6 speed 10g
set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 8 speed 10g
set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 7 speed 25g
set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 9 speed 25g

user@host# delete chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 6
user@host# delete chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 8
[edit]
user@host# show chassis | display set
set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 7 speed 25g
set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 9 speed 25g
user@host# show chassis
fpc 1 {
    pic 0 {
        port-profile A-1X400G-1X100G-4X50G {
            port-num 7 {
                speed 25g;
            }
            port-num 9 {
                speed 25g;
            }
        }
    }
}
```

- Delete port profile and configured port speed

```
delete chassis fpc < fpc-slot > pic < pic-slot > port-profile < A-E profile >
```

```
user@host# show chassis | display set
set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 7 speed 25g
set chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G port-num 9 speed 25g

user@host# delete chassis fpc 1 pic 0 port-profile A-1X400G-1X100G-4X50G

user@host# show chassis /* delete profile and ports, speed under the port-profile */
```

- To show the current port profile:

  ```
  show chassis pic port-profile fpc-slot <1> pic-slot < 0/1 >
  ```

  ```
  user@host# show chassis fpc 1 pic 1 | display set
  set chassis fpc 1 pic 1 port-profile A-1X400G-1X100G-4X50G port-num 6 speed 1g
  set chassis fpc 1 pic 1 port-profile A-1X400G-1X100G-4X50G port-num 8 speed 1g
  set chassis fpc 1 pic 1 port-profile A-1X400G-1X100G-4X50G port-num 7 speed 10g
  set chassis fpc 1 pic 1 port-profile A-1X400G-1X100G-4X50G port-num 9 speed 10g
  ```

  ```
  user@host> show chassis pic port-profile fpc-slot 1 pic-slot 1
  Port profile speed information:
  Current port profile: A-1X400G-1X100G-4X50G
  Port Current Speed Default Speed Configurable Speed
  0 400GE 400GE 40GE, 100GE, 400GE
  2 100GE 100GE 100GE
  6 1GE 50GE 1GE, 10GE, 25GE, 50GE*
  7 10GE 50GE 1GE, 10GE, 25GE, 50GE*
  8 1GE 50GE 1GE, 10GE, 25GE, 50GE*
  9 10GE 50GE 1GE, 10GE, 25GE, 50GE*
  * Note: Paired ports (6,8), (7,9) must be configured to the same speed.
  ```

Follow the guidelines below while configuring the port profile mode on a PIC:

- Starting in 24.4R1-S3, restart chassis-control using the operational mode command `restart chassis-control` as soon as you configure, delete, or switch to the port profile or PIC mode on a PIC.
- You can configure only one port profile on a PIC at a time.
- You need not reboot or restart chassis-control when a port speed is configured, switched, or deleted within a port profile.
- PIC and port profile modes are mutually exclusive. You cannot configure both PIC mode and port profile mode on a PIC simultaneously.

4x10G Channelization on SRX4700 Firewall

SRX4700 firewall supports channelization on port 0 in port-profile configuration D. Channelization support is with a fixed configuration of 1-4 subport range.

Note the following about port channelization on port 0 of 1x400G/4x100G/8x50G PIC:

- 4x10G port channelization is supported only on port 0 in port-profile D.
- Port channelization is supported with a fixed configuration of range 1-4 subports.
- Perform a single commit for:
  - Configuration or deletion of port 0 to 4x10G, 3x10G, 2x10G, or 1x10G.
  - Configuration or deletion of port unused port on port 1.

Use the following command to configure channelization:

- To configure the port-profile and specific port speed and number of subports:

  ```
  set chassis fpc < fpc-slot > pic < pic-slot > port-profile < D profile > port-num <port-number> speed <port-speed> number-of-subports < number-of-sub-ports>
  ```

- To configure the port-profile and port unused:

  ```
  set chassis fpc < fpc-slot > pic < pic-slot > port-profile < D profile > port <port-number> unused
  ```

- To delete the port-profile and its configured port speed:

  ```
  delete chassis fpc < fpc-slot > pic < pic-slot > port-profile < D profile > port <port-number>
  ```

Example:

- Port profile configuration

```
root@srx4700> show chassis pic port-profile fpc-slot 1 pic-slot 0
Port profile speed information:
Current port profile: D-3X100G-8X50G
Port    Current Speed     Default Speed     Configurable Speed
0       10GE              100GE             4x10GE*, 100GE
1       Unused            100GE             100GE
2       100GE             100GE             100GE
6       50GE              100GE             1GE, 10GE, 25GE, 50GE*
7       50GE              100GE             1GE, 10GE, 25GE, 50GE*
8       50GE              100GE             1GE, 10GE, 25GE, 50GE*
9       50GE              100GE             1GE, 10GE, 25GE, 50GE*
10      50GE              100GE             1GE, 10GE, 25GE, 50GE*
11      50GE              100GE             1GE, 10GE, 25GE, 50GE*
12      50GE              100GE             1GE, 10GE, 25GE, 50GE*
13      50GE              100GE             1GE, 10GE, 25GE, 50GE*
```

Paired ports (6,8), (7,9), (10,12), (11,13) must be configured to the same speed.

- 4x10G port channelization

```
root@srx4700 # set chassis fpc 1 pic 0 port-profile D-3X100G-8X50G port-num 1
unused
root@srx4700 # set chassis fpc 1 pic 0 port-profile D-3X100G-8X50G port-num 0 speed 10g
number-of-sub-ports 4
root@srx4700 # commit
```

```
root@srx4700# show
chassis
fpc 1 {
    pic 0 {
        port-profile D-3X100G-8X50G {
            port-num 1 {
                unused;
            }
            port-num 0 {
                speed 10g;
                number-of-sub-ports 4;
            }
        }
    }
```

- Configuration deletion

```
root@srx4700 # delete chassis fpc 1 pic 0 port-profile D-3X100G-8X50G port-num 0
root@srx4700 # delete chassis fpc 1 pic 0 port-profile D-3X100G-8X50G port-num 1
root@srx4700 #commit
root@srx4700 # show chassis
```

Port Speed on SRX5K-IOC4-MRATE

IN THIS SECTION

Each of the twelve ports of PIC 0 and PIC 1 of an SRX5K-IOC4-MRATE supports port speeds of 10 Gbps and 40 Gbps. However, only ports 2 and 5 of both the PICs support port speed of 100 Gbps. You can choose to configure all supported ports of the SRX5K-IOC4-MRATE to operate at the same supported speed or configure all the ports at different supported speeds.

You can configure port speed at the PIC level, to operate all the ports of the SRX5K-IOC4-MRATE at the same speed. That is, you can choose to configure the PIC to operate at a supported speed, and then all the supported ports of the PIC to operate at the configured speed. For example, if you choose to configure PIC 0 at 100-Gbps speed, only ports 2 and 5 of PIC 0 operate at 100-Gbps speed, while the other ports of the PIC are disabled. Similarly, if you choose to configure PIC 0 at 10-Gbps or 40-Gbps speed, all the ports of the PIC are enabled to operate at those speeds. Additionally, you can prevent oversubscription by specifying the number of active physical ports that operate at 10-Gbps, 40-Gbps, and 100-Gbps speeds.

You cannot configure 1-Gbps speed at PIC level and port level. You can configure the port that is configured at 10-Gbps speed to operate at 1-Gbps speed by using the speed statement. After you commit the configuration, the operating speed of the 10-Gbps port changes to 1-Gbps speed, but the show interface command displays the speed configuration (operating port speed) as 1 GbE. If you configure the interface with 1-Gbps speed, then the Speed Configuration field displays 1 GbE; if you configure the interface with 10-Gbps speed, Speed Configuration displays AUTO.

You can configure port speed at the port level, to operate different ports of the SRX5K-IOC4-MRATE at different supported speeds. That is, you configure each port to operate at a supported speed.

The SRX5K-IOC4-MRATE supports an aggregate bandwidth of 480 Gbps, and each of the two PICs supports a bandwidth limit of 240 Gbps. If the aggregate port capacity configured exceeds 240 Gbps per PIC, the configuration is not supported.

Configuring Port Speed at PIC Level

To configure port speed at the PIC level:

1. In configuration mode, navigate to the `[edit chassis fpc fpc-slot pic pic-number]` hierarchy level.

```
[edit ]
user@host# edit chassis fpc fpc-slot pic pic-number
```

For example:

```
[edit ]
user@host# edit chassis fpc 4 pic 0
```

2. Configure the `pic-mode` statement to set the operating speed for the PIC's ports. According to your requirements, you can choose from the options `10G`, `40G`, or `100G`.

```
[edit chassis fpc fpc-slot pic pic-number]
user@host# set pic-mode pic-speed
```

For example:

```
[edit chassis fpc 4 pic 0]
user@host# set pic-mode 10G
```

3. (Optional) To prevent oversubscription, you can choose to configure the number of ports that operate at the mode that is configured in Step 2.

```
[edit chassis fpc fpc-slot pic pic-number]
user@host# set number-of-ports number-of-active-physical-ports
```

For example:

```
[edit chassis fpc 4 pic 0]
user@host# set number-of-ports 6
```

4. Verify the configuration.

```
[edit chassis fpc 4 pic 0]
user@host# show
pic-mode 10G;
number-of-ports 6;
```

5. Commit your configuration changes.

If the `number-of-ports` statement is *not* configured, all the ports that support the speed configured in Step 2 are enabled. That is, depending on that selection, ports 0 through 5 are enabled for speeds of 10 GbE or 40 GbE, while ports 2 and 5 are enabled for 100 GbE. You can also use the `number-of-ports` statement

to disable certain ports. Table 1 below, lists the physical ports that are enabled when the `number-of-ports` statement is configured.

Table 42: Active Physical Ports on SRX5K-IOC4-MRATE Based on the number-of-ports Configuration

Ports Configured (`number-of-ports` Statement)	Active Physical Ports for Different Configured Speeds		
	10-Gigabit	40-Gigabit	100-Gigabit
1	0	0	2
2	0, 1	0, 1	2, 5
3	0, 1, 2	0, 1, 2	2, 5
4	0, 1, 2, 3	0, 1, 2, 3	2, 5
5	0, 1, 2, 3, 4	0, 1, 2, 3, 4	2, 5
6	0, 1, 2, 3, 4, 5	0, 1, 2, 3, 4, 5	2, 5

Configuring Port Speed at Port Level

To configure port speed at the port level:

1. In configuration mode, navigate to the `[edit chassis fpc` *`fpc-slot`* `pic` *`pic-number`*`]` hierarchy level.

```
[edit]
user@host# edit chassis fpc fpc-slot pic pic-number
```

For example:

```
[edit]
user@host# edit chassis fpc 4 pic 0
```

2. To indicate the speed at which the ports operate, configure the `speed` statement for the desired ports. According to your requirements, you can choose the `10g`, `40g`, or `100g` speed options.

```
[edit chassis fpc fpc-slot pic pic-number]
user@host# set port port-number speed (10g | 40g | 100g)
```

For example:

```
[edit chassis fpc 4 pic 0]
user@host# set port 0 speed 10g
user@host# set port 1 speed 10g
user@host# set port 2 speed 100g
user@host# set port 3 speed 40g
```

All the twelve ports of PIC 0 and PIC 1 of an SRX5K-IOC4-MRATE support 10-Gbps and 40-Gbps port speeds. However, only ports 2 and 5 of both the PICs support 100-Gbps speed.

3. Verify the configuration.

```
[edit chassis fpc 4 pic 0]
user@host# show
port 0 {
    speed 10g;
}
port 1 {
    speed 10g;
}
port 2 {
    speed 100g;
}
port 3 {
    speed 40g;
}
```

You can verify the 40-Gbps and 100-Gbps ports configured as 10-Gbps by using the `show interfaces terse` command.

```
[edit chassis fpc 10 pic 0 port 0]
+     speed 10g;

user@host# show interfaces terse
```

```
..
xe-10/0/0:0             up    down
xe-10/0/0:1             up    down
xe-10/0/0:2             up    down
xe-10/0/0:3             up    down
```

4. Commit your configuration changes.

Note the following when configuring port speed on an SRX5K-IOC4-MRATE:

- If port speed is not configured, all ports of the SRX5K-IOC4-MRATE operate as four 10-Gigabit Ethernet interfaces by default. Therefore, when booting the MPC:
  - If port speed is not configured or if invalid port speeds are configured, each port operates as four 10-Gigabit Ethernet interfaces. An alarm is generated to indicate that the ports of the SRX5K-IOC4-MRATE are operating as four 10-Gigabit Ethernet interfaces.
  - If valid port speeds are configured, the MPC PICs operate at the configured speed.
- • When you change an existing port speed configuration at the port level, you must reset the SRX5K-IOC4-MRATE PIC for the configuration to take effect. An alarm is generated indicating the change in port speed configuration.
- • When you change an existing port speed configuration with an *invalid* port speed configuration, an alarm is generated indicating that the port speed configuration is invalid. The MPC continues to operate using the previously configured valid port speed configuration. However, if the MPC or PIC is restarted with the committed invalid port configuration, all ports of the MPC operate as four 10-Gigabit Ethernet interfaces by default.
- • You cannot configure port speed at the PIC level and the port level simultaneously. Error messages are displayed when you try to commit such configurations.
- • When you configure port speed at the port level, only the configured ports are enabled. Other ports are disabled.
- Logical interfaces can be created only on ports that are enabled.
- You must restart the chassis when you change the port profile configuration.

Targeted Broadcast

IN THIS SECTION

Targeted broadcast helps in remote administration tasks such as backups and wake-on LAN (WOL) on a LAN interface, and supports virtual routing and forwarding (VRF) instances. The below topic discuss the process and functioning of targeted broadcast, its configuration details, and the status of the broadcast on various platforms.

Overview

IN THIS SECTION

Targeted broadcast is a process of flooding a target subnet with L3 broadcast IP packets originating from a different subnet. The intent of targeted broadcast is to flood the target subnet with the broadcast packets on a LAN interface without broadcasting to the entire network.

IP directed broadcast is a technique where a broadcast packet is sent to a specific remote subnet, and then broadcast within that subnet. You can use IP directed broadcast to facilitate remote network management by sending broadcast packets to hosts on a specified subnet without broadcasting to the entire network. IP directed broadcast packets are broadcast on only the target subnet. The rest of the network treats IP directed broadcast packets as unicast packets and forwards them accordingly.

Targeted broadcast is configured with various options on the egress interface of the router or switch, and the IP packets are broadcast only on the LAN (egress) interface. Targeted broadcast helps you implement remote administration tasks, such as backups and wake-on LAN (WOL) on a LAN interface, and supports VRF instances.

Regular L3 broadcast IP packets originating from a subnet are broadcast within the same subnet. When these IP packets reach a different subnet, the packets are forwarded to the Routing Engine (to be forwarded to other applications). Hence, remote administration tasks such as backups cannot be performed on a particular subnet through another subnet. As a workaround, you can enable targeted broadcast to forward broadcast packets that originate from a different subnet.

L3 broadcast IP packets have a destination IP address that is a valid broadcast address for the target subnet. These IP packets traverse the network in the same way as unicast IP packets until the packets reach the destination subnet, as follows:

1. In the destination subnet, if the receiving router has targeted broadcast enabled on the egress interface, the IP packets are forwarded to an egress interface and the Routing Engine or to an egress interface only.
2. The IP packets are then translated into broadcast IP packets, which flood the target subnet only through the LAN interface, and all hosts on the target subnet receive the IP packets. The packets are discarded If no LAN interface exists.
3. The final step in the sequence depends on targeted broadcast:
   - If targeted broadcast is not enabled on the receiving router, the IP packets are treated as regular Layer 3 broadcast IP packets and are forwarded to the Routing Engine.
   - If targeted broadcast is enabled without any options, the IP packets are forwarded to the Routing Engine.

You can configure targeted broadcast to forward the IP packets only to an egress interface. The forwarding is helpful when the router is flooded with packets to process, or to both an egress interface and the Routing Engine.

Any *firewall filter* that is configured on the Routing Engine lo0 cannot be applied to IP packets that are forwarded to the Routing Engine as a result of a targeted broadcast. The reason is broadcast packets are forwarded as flood next-hop traffic and not as local next-hop traffic. You can apply a firewall filter only to local next-hop routes for traffic directed toward the Routing Engine.

Targeted Broadcast Overview

Targeted broadcast packets have a destination IP address that is a valid broadcast address for the subnet that is the target of the directed broadcast (the target subnet). The intent of a targeted broadcast is to

flood the target subnet with the broadcast packets without broadcasting to the entire network. Targeted broadcast packets cannot originate from the target subnet.

When you send a targeted broadcast packet, as it travels to the target subnet, the network forwards it in the same way as it forwards a unicast packet. When the packet reaches a switch that is directly connected to the target subnet, the switch checks to see whether targeted broadcast is enabled on the interface that is directly connected to the target subnet:

- If targeted broadcast is enabled on that interface, the switch broadcasts the packet on that subnet by rewriting the destination IP address as the configured broadcast IP address for the subnet. The switch converts the packet to a link-layer broadcast packet that every host on the network processes.
- If targeted broadcast is disabled on the interface that is directly connected to the target subnet, the switch drops the packet.

Targeted Broadcast Implementation

You configure targeted broadcast on a per-subnet basis by enabling targeted broadcast on the L3 interface of the subnet's VLAN. When the switch that is connected to that subnet receives a packet that has the subnet's broadcast IP address as the destination address, the switch broadcasts the packet to all hosts on the subnet.

By default, targeted broadcast is disabled.

When to Enable Targeted Broadcast

Targeted broadcast is disabled by default. Enable targeted broadcast when you want to perform remote management or administration services such as backups or WOL tasks on hosts in a subnet that does not have a direct connection to the Internet.

Enabling targeted broadcast on a subnet affects only the hosts within that subnet. Only packets received on the subnet's L3 interface that have the subnet's broadcast IP address as the destination address is flooded on the subnet.

When Not to Enable Targeted Broadcast

Typically, you do not enable targeted broadcast on subnets that have direct connections to the Internet. Disabling targeted broadcast on a subnet's L3 interface affects only that subnet. If you disable targeted broadcast on a subnet and a packet that has the broadcast IP address of that subnet arrives at the switch, the switch drops the broadcast packet.

If a subnet has a direct connection to the Internet, enabling targeted broadcast on it increases the network's susceptibility to DoS attacks.

A malicious attacker can spoof a source IP address to deceive a network into identifying the attacker as legitimate. The attacker can then send targeted broadcasts with ICMP echo (ping) packets. When the hosts on the network with targeted broadcast enabled receive the ICMP echo packets, the hosts send replies to the victim that has the spoofed source IP address. The replies create a flood of ping replies in a DoS attack that can overwhelm the spoofed source address known as a *smurf* attack. Another common DoS attack on exposed networks with targeted broadcast enabled is a *fraggle* attack. The attack is similar to a smurf attack except that the malicious packet is a UDP echo packet instead of an ICMP echo packet.

Understand Targeted Broadcast

IN THIS SECTION

When packets reach the destination subnet and targeted broadcast is enabled on the receiving switch, the switch converts the targeted broadcast packet into a broadcast. The conversion floods the packet on the target subnet. All hosts on the target subnet receive the targeted broadcast packet.

This topic covers:

Targeted Broadcast Overview

Targeted broadcast packets have a destination IP address that is a valid broadcast address for the subnet that is the target of the directed broadcast (the target subnet). The intent of a targeted broadcast is to flood the target subnet with the broadcast packets without broadcasting to the entire network. Targeted broadcast packets cannot originate from the target subnet.

When you send a targeted broadcast packet, as it travels to the target subnet, the network forwards it in the same way as it forwards a unicast packet. When the packet reaches a switch that is directly connected to the target subnet, the switch checks to see whether targeted broadcast is enabled on the interface that is directly connected to the target subnet:

- If targeted broadcast is enabled on that interface, the switch broadcasts the packet on that subnet by rewriting the destination IP address as the configured broadcast IP address for the subnet. The switch converts the packet to a link-layer broadcast packet that every host on the network processes.
- If targeted broadcast is disabled on the interface that is directly connected to the target subnet, the switch drops the packet.

Targeted Broadcast Implementation

You configure targeted broadcast on a per-subnet basis by enabling targeted broadcast on the L3 interface of the subnet's VLAN. When the switch that is connected to that subnet receives a packet that has the subnet's broadcast IP address as the destination address, the switch broadcasts the packet to all hosts on the subnet.

By default, targeted broadcast is disabled.

When to Enable Targeted Broadcast

Targeted broadcast is disabled by default. Enable targeted broadcast when you want to perform remote management or administration services such as backups or WOL tasks on hosts in a subnet that does not have a direct connection to the Internet.

Enabling targeted broadcast on a subnet affects only the hosts within that subnet. Only packets received on the subnet's L3 interface that have the subnet's broadcast IP address as the destination address is flooded on the subnet.

When Not to Enable Targeted Broadcast

Typically, you do not enable targeted broadcast on subnets that have direct connections to the Internet. Disabling targeted broadcast on a subnet's L3 interface affects only that subnet. If you disable targeted broadcast on a subnet and a packet that has the broadcast IP address of that subnet arrives at the switch, the switch drops the broadcast packet.

If a subnet has a direct connection to the Internet, enabling targeted broadcast on it increases the network's susceptibility to DoS attacks.

A malicious attacker can spoof a source IP address to deceive a network into identifying the attacker as legitimate. The attacker can then send targeted broadcasts with ICMP echo (ping) packets. When the hosts on the network with targeted broadcast enabled receive the ICMP echo packets, the hosts send replies to the victim that has the spoofed source IP address. The replies create a flood of ping replies in a DoS attack that can overwhelm the spoofed source address known as a *smurf* attack. Another common DoS attack on exposed networks with targeted broadcast enabled is a *fraggle* attack. The attack is

similar to a smurf attack except that the malicious packet is a UDP echo packet instead of an ICMP echo packet.

Configure Targeted Broadcast

IN THIS SECTION

Configure Targeted Broadcast

You can configure targeted broadcast on an egress interface with different options.

Either of these configurations is acceptable:

- You can allow the IP broadcast packets destined for a Layer 3 address to be forwarded through the egress interface and to send a copy of the IP broadcast packets to the Routing Engine.
- You can allow the IP broadcast packets to be forwarded through the egress interface only.

Note that the packets are broadcast only if the egress interface is a LAN interface.

To configure targeted broadcast and its options:

1. Configure the interface.

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

or

```
[edit]
user@host# set interfaces irb
```

2. Configure the logical unit number at the [edit interfaces *interface-name*] hierarchy level.

```
[edit interfaces interface-name]
user@host# set unit logical-unit-number
```

3. Configure the protocol family as inet at the [edit interfaces *interface-name* unit *interface-unit-number*] hierarchy level.

```
[edit interfaces interface-name unit interface-unit-number]
user@host# set family inet
```

4. Configure targeted broadcast at the [edit interfaces *interface-name* unit *interface-unit-number* family inet] hierarchy level.

```
[edit interfaces interface-name unit interface-unit-number family inet]
user@host# set targeted-broadcast
```

5. Forward IP broadcast packets to a Layer 3 address:

   a. through the egress interface and send a copy of the same packets to the Routing Engine.

```
[edit interfaces interface-name unit interface-unit-number family inet targeted-broadcast]
user@host# forward-and-send-to-re;
```

   or

   b. through the egress interface only.

```
[edit interfaces interface-name unit interface-unit-number family inet targeted-broadcast]
user@host# forward-only;
```

Display Targeted Broadcast Configuration Options

IN THIS SECTION

The following example topics display targeted broadcast configuration options:

Forward IP Broadcast Packets on the Egress Interface and to the Routing Engine

IN THIS SECTION

Purpose

Display the configuration when targeted broadcast is configured on the egress interface to forward the IP broadcast packets on the egress interface and to send a copy of the same packets to the Routing Engine.

Action

To display the configuration, run the `show` command at the `[edit interfaces` *`interface-name`* `unit` *`interface-unit-number`* `family inet]` where the interface name is ge-2/0/0, the unit value is set to 0, and the protocol family is set to inet.

```
[edit interfaces interface-name unit interface-unit-number family inet]
user@host#show
targeted-broadcast {
    forward-and-send-to-re;
}
```

To display the configuration for irb, run the `show` command at the `[edit interfaces irb unit` *`interface-unit-number`* `family inet]`.

```
[edit interfaces irb unit interface-unit-number family inet]
user@host#show
targeted-broadcast {
    forward-and-send-to-re;
}
```

Forward IP Broadcast Packets on the Egress Interface Only

IN THIS SECTION

Purpose

Display the configuration when targeted broadcast is configured on the egress interface to forward the IP broadcast packets on the egress interface only.

Action

To display the configuration, run the `show` command at the `[edit interfaces` *`interface-name`* `unit` *`interface-unit-number`* `family inet]` where the interface name is ge-2/0/0, the unit value is set to 0, and the protocol family is set to inet.

```
[edit interfaces interface-name unit interface-unit-number family inet]
user@host#show
targeted-broadcast {
    forward-only;
}
```

To display the configuration, run the `show` command at the `[edit interfaces irb unit` *`interface-unit-number`* `family inet]`.

```
[edit interfaces irb unit interface-unit-number family inet]
user@host#show
targeted-broadcast {
    forward-only;
}
```

Power over Ethernet

IN THIS SECTION

Power over Ethernet (PoE) is the implementation of the IEEE 802.3 AF and IEEE 802.3 AT standards that allow both data and electrical power to pass over a copper Ethernet LAN cable. The topics below discuss the overview and configuration details of PoE, and disabling a PoE interface on security devices.

Power over Ethernet Overview

IN THIS SECTION

Power over Ethernet (PoE) is the implementation of the IEEE 802.3 AF and IEEE 802.3 AT standards that allow both data and electrical power to pass over a copper Ethernet LAN cable.

You can configure the SRX Series Firewall to act as power sourcing equipment (PSE), supplying power to powered devices that are connected on designated ports. For more information about PoE, see Power over Ethernet (PoE) User Guide for EX Series Switches.

This topic contains the following sections:

SRX Series Services Gateway PoE Specifications

Table 43 on page 257 lists the PoE specifications for the SRX210, SRX220, SRX240, SRX320, SRX650, and SRX550 M devices. (Platform support depends on the Junos OS release in your installation.)

Table 43: PoE Specifications for the SRX210, SRX220, SRX240, SRX320, and SRX650 Devices

Specifications	For SRX210 Device	For SRX220 Device	For SRX240 Device	For SRX320 PoE Device		For SRX 550 M device	For SRX650 Device
Supported standards	• IEEE 802.3 AF • Legacy (pre-standards)	• IEEE 802.3 AF • IEEE 802.3 AT (PoE +) • Legacy (pre-standards)	• IEEE 802.3 AF • IEEE 802.3 AT (PoE +) • Legacy (pre-standards)	• IEEE 802.3 AF • IEEE 802.3 AT (PoE) • Legacy (pre-standards)		• IEEE 802.3 AF • IEEE 802.3 AT (PoE +) • Legacy (pre-standards)	• IEEE 802.3 AF • IEEE 802.3 AT (PoE +) • Legacy (pre-standards)

Table 43: PoE Specifications for the SRX210, SRX220, SRX240, SRX320, and SRX650 Devices *(Continued)*

Specifications	For SRX210 Device	For SRX220 Device	For SRX240 Device	For SRX320 PoE Device		For SRX 550 M device	For SRX650 Device

Supported ports	Supported on two Gigabit Ethernet ports and two Fast Ethernet ports (ge-0/0/0, ge-0/0/1, fe-0/0/2, and fe-0/0/3).	Supported on all 8 Gigabit Ethernet ports (ge-0/0/0 to ge-0/0/7).	Supported on all 16 Gigabit Ethernet ports (ge-0/0/0 to ge-0/0/15).	Supported on all 6 Copper (RJ45) Gigabit Ethernet ports (ge-0/0/0 to ge-0/0/5).		Supported on 16GE-POE xPIM card	Supported on the following ports:
							• Slot 2 or 6 on 16 Gigabit Ethernet ports
							  • ge-2/0/0 to ge-2/0/15
							  • ge-6/0/0 to ge-6/0/15
							• Slot 2 or 6 on 24 Gigabit Ethernet ports
							  • ge-2/0/0 to ge-2/0/23

Table 43: PoE Specifications for the SRX210, SRX220, SRX240, SRX320, and SRX650 Devices *(Continued)*

Specifications	For SRX210 Device	For SRX220 Device	For SRX240 Device	For SRX320 PoE Device		For SRX 550 M device	For SRX650 Device
							• ge-6/0/0 to ge-6/0/23

Table 43: PoE Specifications for the SRX210, SRX220, SRX240, SRX320, and SRX650 Devices *(Continued)*

Specifications	For SRX210 Device	For SRX220 Device	For SRX240 Device	For SRX320 PoE Device		For SRX 550 M device	For SRX650 Device
Total PoE power sourcing capacity	50 W	120 W	150 W	180 W		The 645 watts AC and 645 watts DC power supplies support the following capacities: • 250 watts on a single power supply, or with redundancy using the two-power-supply option. • 500 watts with the two-power-supply option operating as nonredundant.	The 645 watts AC and 645 watts DC power supplies support the following capacities: • 250 watts on a single power supply, or with redundancy using the two-power-supply option. • 500 watts with the two-power-supply option operating as nonredundant.

Table 43: PoE Specifications for the SRX210, SRX220, SRX240, SRX320, and SRX650 Devices *(Continued)*

Specifications	For SRX210 Device	For SRX220 Device	For SRX240 Device	For SRX320 PoE Device		For SRX 550 M device	For SRX650 Device
Default per port power limit	15.4 W	15.4 W	15.4 W	30 W		15.4 W	15.4 W
Maximum per port power limit	30 W	30W	30 W	30 W		30 W	30 W
Power management modes	• Static: Power allocated for each interface can be configured. • Class: Power allocated for interfaces is based on the class of powered device connected.	• Static: Power allocated for each interface can be configured. • Class: Power allocated for interfaces is based on the class of powered device connected.	• Static: Power allocated for each interface can be configured. • Class: Power allocated for interfaces is based on the class of powered device connected.	• Static: Power allocated for each interface can be configured. • Class: Power allocated for interfaces is based on the class of powered device connected.		• Static: Power allocated for each interface can be configured. • Class: Power allocated for interfaces is based on the class of powered device connected.	• Static: Power allocated for each interface can be configured. • Class: Power allocated for interfaces is based on the class of powered device connected.

PoE Classes and Power Ratings

Table 44 on page 262 lists the classes and their power ratings as specified by the IEEE standards.

Table 44: SRX Series Firewalls PoE Specifications

Class	Usage	Minimum Power Levels Output from PoE Port
0	Default	15.4 W
1	Optional	4.0 W
2	Optional	7.0 W
3	Optional	15.4 W
4	Reserved	Class 4 power devices are eligible to receive power up to 30 W according to IEEE standards.

PoE Options

When you configure PoE, you must enable the PoE interface for the port to provide power to a connected, powered device. In addition, you can configure the following PoE features:

- Port priority—Sets port priority. When it is not possible to maintain power to all connected ports, lower priority ports are powered off before higher priority ports. When you connect a device on a higher-priority port, a lower priority port will be powered off automatically if available power is insufficient to power on the higher priority port. (For the ports with the same priority configuration, ports on the left are given higher priority than the ports on the right.)
- Maximum available wattage power available to a port—Sets the maximum amount of power that can be supplied to the port. Default wattage per port is 15.4 watts.
- PoE power consumption logging—Allows logging of per-port PoE power consumption. The telemetries section is disabled by default and must be explicitly specified to enable logging. Default telemetry duration is 1 hour, and the interval is 5 minutes.
- PoE power management mode—Has two modes:

- Class—Power is allocated dynamically using the classification process.
- Static—Power is allocated based on the maximum power configuration.

- Reserve power—Specified amount of power is reserved for the gateway in case of a spike in PoE consumption. The default is 0.

Example: Configure PoE Interface

IN THIS SECTION

In this topic you can learn to configure the PoE interface on all interfaces, an individual interface, and how to disable a PoE interface. Below table specifies the CLI quick configuration commands used for configuring PoE interfaces.

CLI Quick Configuration

Use the below table to view the CLI quick configuration commands to configure PoE on individual and all interfaces, and also to disabe the interface.

Table 45: CLI Quick Configuration

Configuration Step	CLI Quick Configuration Commands
Configure PoE on an individual interface.	`set poe interface ge-0/0/0 priority high maximum-power 15.4 telemetries` `set poe management static guard-band 15`
Configure PoE on all individual interfaces.	`set poe interface all priority low maximum-power 15.4 telemetries` `set poe management static guard-band 15`

Configure PoE Interfaces

Below table describes the steps to configure PoE interfaces on your security device.

Table 46: PoE Interfaces Configuration

Configuration Step	Command

Step 1: Enable PoE	For an individual interface:
	
	`[edit`
	`user@host# edit poe interface ge-0/0/0`
	
	For all interfaces:
	
	`[edit]`
	`user@host# edit poe interface all`

Step 2: Set the power port priority.	For an individual interface:
	
	`[edit poe interface ge-0/0/0]`
	`user@host# set priority high`
	
	For all interfaces:
	
	`[edit poe interface all]`
	`user@host# set priority low`

Step 3: Set the maximum PoE wattage available for a port.	For an individual interface:
	
	`[edit poe interface ge-0/0/0]`
	`user@host# set maximum power 15.4`
	
	For all interfaces:
	
	`[edit poe interface all]`
	`user@host# set maximum power 15.4`

Table 46: PoE Interfaces Configuration *(Continued)*

Configuration Step	Command

Step 4: Enable logging of PoE power consumption.	For an individual interface:
	
	`[edit poe interface ge-0/0/0]`
	`user@host# set telemetries`
	
	For all interfaces:
	
	`[edit poe interface all]`
	`user@host# set telemetries`

Set the PoE management mode.	`[edit]` `user@host# set poe management static`
Step 6: Reserve power wattage in case of a spike in PoE consumption.	`[edit]` `user@host# set poe guard-band 15`
Step 7: (Optional) Disable PoE on all interfaces.	`[edit]` `user@host# set poe interface all disable`
Step 8: (Optional) Disable PoE on a specific interface.	`[edit]` `user@host# set poe interface ge-0/0/0 disable`
Step 9: If you are done configuring the device, commit the configuration.	`[edit]` `user@host# commit`

Use the `show poe interface ge-0/0/0` and `show poe interface all` command to see the output of the configuration. To verify the configuration is working properly, enter the `show poe interface`command.

Verification

Purpose

Verify that PoE interface is enabled on individual and all interfaces, also check how to disable PoE interface. (The device used in this example is the SRX240 or SRX340 Firewall, depending on the Junos OS release in the installation.)

Action

- To display information about the parameters configured on PoE interface.

```
user@host> show poe interface ge-0/0/1
PoE interface status:
PoE interface                : ge-0/0/1
Administrative status        : Enabled
Operational status           : Powered-up
Power limit on the interface : 15.4 W
Priority                     : High
Power consumed               : 6.6 W
Class of power device        : 0
```

- Verify the PoE interface's power consumption over a specified period.

  For all records:

```
user@host> show poe telemetries interface ge-0/0/1 all
Sl No Timestamp Power Voltage
1 Fri Jan 04 11:41:15 2009 5.1 W 47.3 V
2 Fri Jan 04 11:40:15 2009 5.1 W 47.3 V
3 Fri Jan 04 11:39:15 2009 5.1 W 47.3 V
4 Fri Jan 04 11:38:15 2009 0.0 W 0.0 V
5 Fri Jan 04 11:37:15 2009 0.0 W 0.0 V
6 Fri Jan 04 11:36:15 2009 6.6 W 47.2 V
7 Fri Jan 04 11:35:15 2009 6.6 W 47.2 V
```

  For a specific number of records:

```
user@host> show poe telemetries interface ge-0/0/1 5
Sl No Timestamp Power Voltage
```

```
1 Fri Jan 04 11:31:15 2009 6.6 W 47.2 V
2 Fri Jan 04 11:30:15 2009 6.6 W 47.2 V
3 Fri Jan 04 11:29:15 2009 6.6 W 47.2 V
4 Fri Jan 04 11:28:15 2009 6.6 W 47.2 V
5 Fri Jan 04 11:27:15 2009 6.6 W 47.2 V
```

The telemetry status displays the power consumption history for the specified interface, provided telemetry has been configured for that interface.

- Verify global parameters such as guard band, power limit, and power consumption.

```
user@host> show poe controller
Controller  Maximum   Power          Guard band  Management
index       power     consumption
   0        150.0 W    0.0 W         0 W         Static
```

- Verify that the PoE interfaces on the device are enabled and set to the desired priority settings. (The device used here is the SRX340 Firewall.)

```
user@host> show poe interface all
```

```
Interface Admin status Oper status Max power Priority Power consumption Class
 ge-0/0/0 Enabled      Searching   15.4W     Low      0.0W                0
 ge-0/0/1 Enabled      Powered-up  15.4W     High     6.6W                0
 ge-0/0/2 Disabled     Disabled    15.4W     Low      0.0W                0
 ge-0/0/3 Disabled     Disabled    15.4W     Low      0.0W                0
```

This output shows that the device has four PoE interfaces of which two are enabled with default values. One port has a device connected that is drawing power within expected limits.

4

CHAPTER

Configuring Interface Encapsulation

IN THIS CHAPTER

Interface Encapsulation Overview

IN THIS SECTION

The below topics discuss the overview of overview of physical encapsulation, frame relay encapsulation, point-to-point protocol and high-level data link control.

Understanding Physical Encapsulation on an Interface

Encapsulation is the process by which a lower level protocol accepts a message from a higher level protocol and places it in the data portion of the lower level frame. As a result, datagrams transmitted through a physical network have a sequence of headers: the first header for the physical network (or Data Link Layer) protocol, the second header for the Network Layer protocol (IP, for example), the third header for the Transport Layer protocol, and so on.

The following encapsulation protocols are supported on physical interfaces:

- Frame Relay Encapsulation. See "Understanding Frame Relay Encapsulation on an Interface" on page 270.
- Point-to-Point Protocol. See "Understanding Point-to-Point Protocol" on page 272.
- Point-to-Point Protocol over Ethernet. See "Understanding Point-to-Point Protocol over Ethernet" on page 303.
- High-Level Data Link Control. See "Understanding High-Level Data Link Control" on page 275.

SEE ALSO

Understanding Interfaces | 2

Understanding Frame Relay Encapsulation on an Interface

IN THIS SECTION

The Frame Relay packet-switching protocol operates at the Physical Layer and Data Link Layer in a network to optimize packet transmissions by creating virtual circuits between hosts. Figure 14 on page 270 shows a typical Frame Relay network.

Figure 14: Frame Relay Network

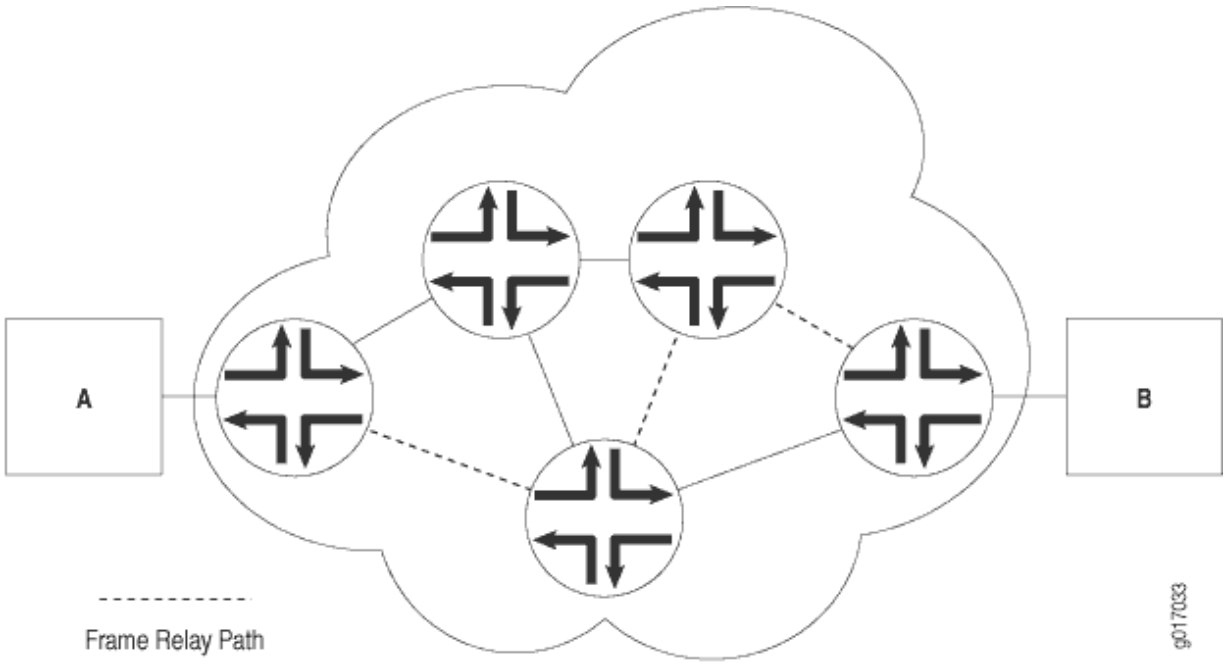


Figure 14 on page 270 shows multiple paths from Host A to Host B. In a typical routed network, traffic is sent from device to device with each device making routing decisions based on its own routing table. In a packet-switched network, the paths are predefined. Devices switch a packet through the network according to predetermined next-hops established when the virtual circuit is set up.

This topic contains the following sections:

Virtual Circuits

A virtual circuit is a bidirectional path between two hosts in a network. Frame Relay virtual circuits are logical connections between two hosts that are established either by a call setup mechanism or by an explicit configuration.

A virtual circuit created through a call setup mechanism is known as a switched virtual circuit (SVC). A virtual circuit created through an explicit configuration is called a permanent virtual circuit (PVC).

Switched and Permanent Virtual Circuits

Before data can be transmitted across an SVC, a signaling protocol like ISDN must set up a call by the exchange of setup messages across the network. When a connection is established, data is transmitted across the SVC. After data transmission, the circuit is torn down and the connection is lost. For additional traffic to pass between the same two hosts, a subsequent SVC must be established, maintained, and terminated.

Because PVCs are explicitly configured, they do not require the setup and teardown of SVCs. Data can be switched across the PVC whenever a host is ready to transmit. SVCs are useful in networks where data transmission is sporadic and a permanent circuit is not needed.

Data-Link Connection Identifiers

An established virtual circuit is identified by a data-link connection identifier (DLCI). The DLCI is a value from 16 through 1022. (Values 1 through 15 are reserved.) The DLCI uniquely identifies a virtual circuit locally so that devices can switch packets to the appropriate next-hop address in the circuit. Multiple paths that pass through the same transit devices have different DLCIs and associated next-hop addresses.

Congestion Control and Discard Eligibility

Frame Relay uses the following types of congestion notification to control traffic within a Frame Relay network. Both are controlled by a single bit in the Frame Relay header.

- Forward explicit congestion notification (FECN)
- Backward explicit congestion notification (BECN)

Traffic congestion is typically defined in the buffer queues on a device. When the queues reach a predefined level of saturation, traffic is determined to be congested. When traffic congestion occurs in a virtual circuit, the device experiencing congestion sets the congestion bits in the Frame Relay header to

1. As a result, transmitted traffic has the FECN bit set to 1, and return traffic on the same virtual circuit has the BECN bit set to 1.

When the FECN and BECN bits are set to 1, they provide a congestion notification to the source and destination devices. The devices can respond in either of two ways: to control traffic on the circuit by sending it through other routes, or to reduce the load on the circuit by discarding packets.

If devices discard packets as a means of congestion (flow) control, Frame Relay uses the discard eligibility (DE) bit to give preference to some packets in discard decisions. A DE value of 1 indicates that the frame is of lower importance than other frames and more likely to be dropped during congestion. Critical data (such as signaling protocol messages) without the DE bit set is less likely to be dropped.

Understanding Point-to-Point Protocol

IN THIS SECTION

The Point-to-Point Protocol (PPP) is an encapsulation protocol for transporting IP traffic across point-to-point links. PPP is made up of three primary components:

- Link Control Protocol (LCP)—Establishes working connections between two points.
- Authentication protocol—Enables secure connections between two points.
- Network control protocol (NCP)—Initializes the PPP protocol stack to handle multiple Network Layer protocols, such as IPv4, IPv6, and Connectionless Network Protocol (CLNP).

This topic contains the following sections:

Link Control Protocol

LCP is responsible for establishing, maintaining, and tearing down a connection between two endpoints. LCP also tests the link and determines whether it is active. LCP establishes a point-to-point connection as follows:

1. LCP must first detect a clocking signal on each endpoint. However, because the clocking signal can be generated by a network clock and shared with devices on the network, the presence of a clocking signal is only a preliminary indication that the link might be functioning.
2. When a clocking signal is detected, a PPP host begins transmitting PPP Configure-Request packets.
3. If the remote endpoint on the point-to-point link receives the Configure-Request packet, it transmits a Configure-Acknowledgement packet to the source of the request.
4. After receiving the acknowledgement, the initiating endpoint identifies the link as established. At the same time, the remote endpoint sends its own request packets and processes the acknowledgement packets. In a functioning network, both endpoints treat the connection as established.

During connection establishment, LCP also negotiates connection parameters such as FCS and HDLC framing. By default, PPP uses a 16-bit FCS, but you can configure PPP to use either a 32-bit FCS or a 0-bit FCS (no FCS). Alternatively, you can enable HDLC encapsulation across the PPP connection.

After a connection is established, PPP hosts generate Echo-Request and Echo-Response packets to maintain a PPP link.

PPP Authentication

PPP's authentication layer uses a protocol to help ensure that the endpoint of a PPP link is a valid device. Authentication protocols include the Password Authentication Protocol (PAP), the Extensible Authentication Protocol (EAP), and the Challenge Handshake Authentication Protocol (CHAP). CHAP is the most commonly used.

NOTE: Support for user id and the password to comply with full ASCII character set is supported through RFC 2486.

The user can enable or disable the RFC 2486 support under the PPP options. The RFC 2486 is disabled by default, and enable the support globally use the command `set access ppp-options compliance rfc 2486`".

CHAP ensures secure connections across PPP links. After a PPP link is established by LCP, the PPP hosts at either end of the link initiate a three-way CHAP handshake. Two separate CHAP handshakes are required before both sides identify the PPP link as established.

CHAP configuration requires each endpoint on a PPP link to use a shared secret (password) to authenticate challenges. The shared secret is never transmitted over the wire. Instead, the hosts on the PPP connection exchange information that enables both to determine that they share the same secret. Challenges consist of a hash function calculated from the secret, a numeric identifier, and a randomly chosen challenge value that changes with each challenge. If the response value matches the challenge value, authentication is successful. Because the secret is never transmitted and is required to calculate the challenge response, CHAP is considered very secure.

PAP authentication protocol uses a simple two-way handshake to establish identity. PAP is used after the link establishment phase (LCP up), during the authentication phase. Junos OS can support PAP in one direction (egress or ingress), and CHAP in the other.

Network Control Protocols

After authentication is completed, the PPP connection is fully established. At this point, any higher level protocols (for example, IP protocols) can initialize and perform their own negotiations and authentication.

PPP NCPs include support for the following protocols. IPCP and IPv6CP are the most widely used on SRX Series Firewalls.

- IPCP—IP Control Protocol
- IPv6CP—IPv6 Control Protocol
- OSINLCP—OSI Network Layer Control Protocol (includes IS-IS, ES-IS, CLNP, and IDRP)

Magic Numbers

Hosts running PPP can create “magic” numbers for diagnosing the health of a connection. A PPP host generates a random 32-bit number and sends it to the remote endpoint during LCP negotiation and echo exchanges.

In a typical network, each host's magic number is different. A magic number mismatch in an LCP message informs a host that the connection is not in loopback mode and traffic is being exchanged bidirectionally. If the magic number in the LCP message is the same as the configured magic number, the host determines that the connection is in loopback mode, with traffic looped back to the transmitting host.

Looping traffic back to the originating host is a valuable way to diagnose network health between the host and the loopback location. To enable loopback testing, telecommunications equipment typically supports channel service unit/data service unit (CSU/DSU) devices.

CSU/DSU Devices

A channel service unit (CSU) connects a terminal to a digital line. A data service unit (DSU) performs protective and diagnostic functions for a telecommunications line. Typically, the two devices are packaged as a single unit. A CSU/DSU device is required for both ends of a T1 or T3 connection, and the units at both ends must be set to the same communications standard.

A CSU/DSU device enables frames sent along a link to be looped back to the originating host. Receipt of the transmitted frames indicates that the link is functioning correctly up to the point of loopback. By configuring CSU/DSU devices to loop back at different points in a connection, network operators can diagnose and troubleshoot individual segments in a circuit.

Understanding High-Level Data Link Control

IN THIS SECTION

High-Level Data Link Control (HDLC) is a bit-oriented, switched and nonswitched link-layer protocol. HDLC is widely used because it supports half-duplex and full-duplex connections, point-to-point and point-to-multipoint networks, and switched and nonswitched channels.

This topic contains the following sections:

HDLC Stations

Nodes within a network running HDLC are called stations. HDLC supports three types of stations for data link control:

- Primary stations—Responsible for controlling the secondary and combined other stations on the link. Depending on the HDLC mode, the primary station is responsible for issuing acknowledgement packets to allow data transmission from secondary stations.
- Secondary stations—Controlled by the primary station. Under normal circumstances, secondary stations cannot control data transmission across the link with the primary station, are active only when requested by the primary station, and can respond to the primary station only (not to other secondary stations). All secondary station frames are response frames.

- Combined stations—A combination of primary and secondary stations. On an HDLC link, all combined stations can send and receive commands and responses without any permission from any other stations on the link and cannot be controlled by any other station.

HDLC Operational Modes

HDLC runs in three separate modes:

- Normal Response Mode (NRM)—The primary station on the HDLC link initiates all information transfers with secondary stations. A secondary station on the link can transmit a response of one or more information frames only when it receives explicit permission from the primary station. When the last frame is transmitted, the secondary station must wait for explicit permission before it can transmit more frames.

  NRM is used most widely for point-to-multipoint links, in which a single primary station controls many secondary stations.

- Asynchronous Response Mode (ARM)—The secondary station can transmit either data or control traffic at any time, without explicit permission from the primary station. The primary station is responsible for error recovery and link setup, but the secondary station can transmit information at any time.

  ARM is used most commonly with point-to-point links, because it reduces the overhead on the link by eliminating the need for control packets.

- Asynchronous Balance Mode (ABM)—All stations are combined stations. Because no other station can control a combined station, all stations can transmit information without explicit permission from any other station. ABM is not a widely used HDLC mode.

Configuring GRE Keepalive Time

IN THIS SECTION

Generic routing encapsulation (GRE) tunnel interfaces do not have a built-in mechanism for detecting when a tunnel is down. Keepalive messages help the GRE tunnel interfaces to detect when a tunnel is down. The topics below discuss the working and configuration of GRE keepalive time.

Understanding GRE Keepalive Time

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.

Keepalive times are only configurable for the ATM-over-ADSL interface, which is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM starting in Junos OS Release 15.1X49-D10. Keepalive times are enabled by default for other interfaces.

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.

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.

SEE ALSO

keepalive-time

hold-time

Configuring GRE Keepalive Time

IN THIS SECTION

Keepalive times are only configurable for the ATM-over-ADSL interface, which is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM starting in Junos OS Release 15.1X49-D10.

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 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

IN THIS SECTION

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

IN THIS SECTION

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.

Example: GRE Configuration

IN THIS SECTION

Generic routing encapsulation (GRE) is an IP encapsulation protocol that is used to transport packets over a network. Information is sent from one network to the other through a GRE tunnel. GRE encapsulates a payload as a GRE packet. This GRE packet is encapsulated in an outer protocol (delivery protocol). GRE tunnel endpoints forward payloads into GRE tunnels for routing packets to the destination. After reaching the end point, GRE encapsulation is removed and the payload is transmitted to its final destination. The primary use of GRE is to carry non-IP packets through an IP network; however, GRE is also used to carry IP packets through an IP cloud.

Requirements

- Configure a GRE (gr-) interface. The gr- interface contains a local address and destination address. It comes up as soon as it is configured. You can even configure an IP address on the gr- interface.
- Configure a route to reach the destination subnet (end-to-end connectivity). You can configure either a static route through the gr- interface or use an interior gateway protocol (IGP) such as OSPF.

Overview

GRE tunnels are designed to be completely stateless, which means that each tunnel endpoint does not keep any information about the state or availability of the remote tunnel endpoint. Normally, a GRE tunnel interface comes up as soon as it is configured, and it stays up as long as there is a valid tunnel source address or interface that is up.

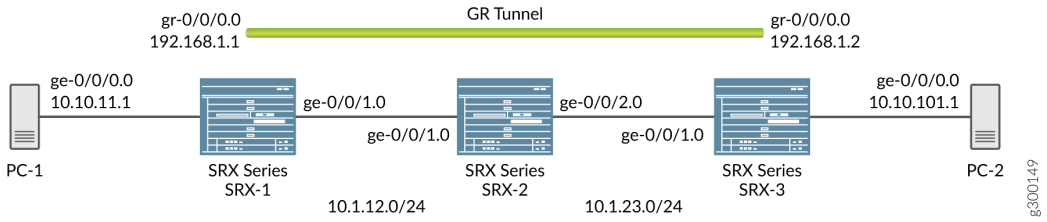
Configuration

IN THIS SECTION

By default, the local subnet interface is ge-0/0/0 with IPv4 address as 10.10.11.1/24. The destination subnet is 10.10.10.0/24 with the tunnel endpoint IPv4 interface as 10.10.10.1/24.

GRE configuration shows the default configuration between the tunnel interfaces on SRX Series Firewalls.

Figure 15: GRE Configuration



Configuring a Route to Reach the Destination Subset

Step-by-Step Procedure

You can either configure a static route through the gr- interface or by using IGP.

1. Configure the local subnet interface ge-0/0/0 interface.

```
[edit interfaces]
user@host# set interfaces ge-0/0/0 unit 0 family inet address 10.10.11.1/24
```

2. Configure the interface ge-0/0/1.

```
[edit interfaces]
user@host# set interfaces ge-0/0/1 unit 0 family inet address 10.1.12.1/24
```

3. Configure the gr- tunnel endpoints and specify the source address, destination address, and family as inet for the tunnel endpoints.

```
[edit interfaces]
user@host# set interfaces gr-0/0/0  unit 0 tunnel source 10.1.12.1 destination 10.1.23.1
user@host# set interfaces gr-0/0/0 unit 0 family inet address 192.168.1.1/24
```

4. The configured interfaces are bound to a security zone at the `[edit security]` hierarchy level. Use the `show zones` command to view the zones. Configure the zones as follows:

```
[edit security zones security-zones trust]]
user@host# set host-inbound-traffic system-services all
user@host# set host-inbound-traffic protocols all
user@host# set interfaces gr-0/0/0.0
user@host# set zones zone names protocols all
```

```
[edit security zones security-zones untrust]]
user@host# set host-inbound-traffic system-services all
user@host# set host-inbound-traffic protocols all
```

5. View the configured interfaces at the `[edit interfaces]` hierarchy level using the `show` command.

```
[edit interfaces]
user@host# set routing options static route 10.10.10.0/24 next hop gr-0/0/0.0
```

6. In case you do not want to define a static route, OSPF can be configured between gr-0/0/0 interfaces on both the sides and internal subnet as passive neighbor, to receive all the internal routes. Configure OSPF at the `[edit protocols]` hierarchy level and view it using the `show` command.

```
[edit protocols]
user@host# set protocols ospf area 0.0.0.0 interface gr-0/0/0.0
```

Results

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

GRE configuration using the static route:

```
[edit interfaces]
root@SRX-1# show
ge-0/0/0 {
    unit 0 {
        family inet {
```

```
            address 10.10.11.1/24;
        }
    }
}

gr-0/0/0 {
    unit 0 {
        tunnel {
            source 10.1.12.1;
            destination 10.1.23.1;
        }
        family inet {
            address 192.168.1.1/24;
        }
    }
}

ge-0/0/1 {
    unit 0 {
        family inet {
            address 10.1.12.1/24;
        }
    }
}

[edit security]
root@SRX-1# show
zones {
    security-zone trust {
        host-inbound-traffic {
            system-services {
                all;
            }
            protocols {
                all;
            }
        }
        interfaces {
            gr-0/0/0.0;
        }
    }
}
```

```
root@SRX-1# show routing-options
static {
    route 10.10.10.0/24 next-hop gr-0/0/0.0;
}
```

GRE configuration using OSPF configured between interfaces gr-0/0/0 on both sides and internal subnet as passive neighbor:

```
[edit protocols]
root@SRX-1# show
ospf {
    area 0.0.0.0 {
        interface gr-0/0/0.0;
        interface ge-0/0/0.0 {
            passive;
        }
    }
}
```

Verification

IN THIS SECTION

To verify that the configuration of GRE on the SRX Series Firewall is successful, perform the following tasks:

Verification of the GRE Interfaces

Purpose

Verify that the GRE interfaces are up.

Action

Run the `show interfaces` command at the `[edit interfaces]` hierarchy level:

```
show interfaces gr-0/0/0 terse
[edit interfaces]
Interface Admin Link Proto Local Remote
gr-0/0/0   up up
gr-0/0/0.0 up up inet 192.168.1.1/24
```

Verification of the Route

Purpose

Verify that the route for the destination network is reachable through the GRE tunnel interface.

Action

Run the `show route forwarding-table matching` *`10.10.10.0/24`* command at the `[edit interfaces]` hierarchy level:

```
[edit interfaces]
user@router# run show route forwarding-table matching 10.10.10.0/24
Routing table: default.inet
Internet:
 ....
Destination        Type RtRef Next hop           Type Index    NhRef Netif
10.10.10.0/24      user     0                    ucst      595     2 gr-0/0/0.0
```

Verification of Traffic Through GRE Tunnel

Purpose

Send the traffic to the destination subnet and verify when the GRE interface is up.

Action

Run the `show interfaces gr-0/0/0 extensive` operational command. Also verify that the packets are leaving through the gr- interface.

```
user@host> show interfaces gr-0/0/0 extensive
Physical interface: gr-0/0/0, Enabled, Physical link is Up
Interface index: 134, SNMP ifIndex: 40, Generation: 17
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: 2005-08-05 21:39:41 UTC (00:00:47 ago)
Traffic statistics:
Input bytes : 8400 0 bps
Output bytes : 8400 0 bps
Input packets: 100 0 pps
Output packets: 100 0 pps

Logical interface gr-0/0/0.0 (Index 72) (SNMP ifIndex 28) (Generation 17)
Flags: Point-To-Point SNMP-Traps 16384
IP-Header 10.1.12.1:10.1.23.1:47:df:64:0000000000000000
Encapsulation: GRE-NULL
Traffic statistics:
Input bytes : 8400
Output bytes : 8400
Input packets: 100
Output packets: 100
Local statistics:
Input bytes : 0
Output bytes : 0
Input packets: 0
Output packets: 0
Transit statistics:
Input bytes : 8400 0 bps
Output bytes : 8400 0 bps
Input packets: 100 0 pps
Output packets: 100 0 pps
Protocol inet, MTU: 1476, Generation: 25, Route table: 0
Flags: None
Addresses, Flags: Is-Primary
```

```
Destination: Unspecified, Local: 192.168.0.1, Broadcast: Unspecified,
Generation: 30
```

SEE ALSO

Generic Routing Encapsulation (GRE)

Understanding Generic Routing Encapsulation

Verifying That Generic Routing Encapsulation Tunneling Is Working Correctly

Example: Configuring GRE over IPsec Tunnels

IN THIS SECTION

Requirements

Overview

GRE tunnels offer minimal security, whereas an IPsec tunnel offers enhanced security in terms of confidentiality, data authentication, and integrity assurance. Also, IPsec cannot directly support multicast packets. However, if an encapsulated GRE tunnel is used first, an IPsec tunnel can then be used to provide security to the multicast packet. In a GRE over IPsec tunnel, all of the routing traffic (IP and non-IP) can be routed through. When the original packet (IP/non-IP) is GRE encapsulated, it has an IP header as defined by the GRE tunnel, normally the tunnel interface IP addresses. The IPsec protocol can understand the IP packet; so it encapsulates the GRE packet to make it GRE over IPsec.

The basic steps involved in configuring GRE over IPsec are as follows:

- Configure the route-based IPsec tunnel.
- Configure the GRE tunnel.

- Configure a static route with the destination as the remote subnet through the gr- interface.
- Configure the static route for the GRE endpoint with the st0 interface as next hop.

Configuration

IN THIS SECTION

In this example, the default configuration has the local subnet interface as ge-0/0/0 with the IPv4 address as 10.10.11.1/24. The destination subnet is 10.10.10.0/24. The gr-0/0/0 interface tunnel endpoints are loopback addresses on both the sides, with the local loopback IPv4 address as 172.20.1.1 and the remote loopback IPv4 address as 172.20.1.2. The gr-0/0/0, st0 and lo0 interfaces are bound to a security zone and policies are created accordingly.

Configuring a GRE interface over an IPsec tunnel

Step-by-Step Procedure

1. Configure the GRE at the [set interfaces *interface-name* unit *unit-number*] hierarchy level, where the interface name is ge-0/0/0, and the family is set as inet.

```
[edit interfaces]
user@host# set interfaces ge-0/0/0 unit 0 family inet address 10.10.11.1/24
```

2. Configure the gr- tunnel endpoints and specify the source address, destination address, and family as inet for the tunnel endpoints.

```
[edit interfaces]
user@host# set interfaces gr-0/0/0  unit 0 tunnel source 172.20.1.1 destination 172.20.1.2
user@host# set interfaces gr-0/0/0  unit 0 family inet 192.168.1.1/24
```

3. Similarly configure the lo0 and st0 interface with the family set as inet.

```
[edit interfaces]
user@host# set interfaces lo0 unit 0 family inet address 172.20.1.1/32
```

```
[edit interfaces]
user@host# set interfaces st0 unit 0 family inet
```

4. Confifure the GRE interfaces with security zones. Use the `show zones` command to view the zones, where the configured tunnel interfaces, lo0 and st0 are displayed.

```
[edit security zones security-zones trust]]
user@host# set host-inbound-traffic system-services all
user@host# set host-inbound-traffic protocols all
user@host# set interfaces gr-0/0/0.0
user@host# set zones zone names protocols all
user@host# set interfaces lo0.0
user@host# set interfaces st0.0
```

```
[edit security zones security-zones untrust]]
user@host# set host-inbound-traffic system-services all
user@host# set host-inbound-traffic protocols all
user@host# set interfaces gr-0/0/0.0.1
user@host# set interfaces lo0.0
user@host# set interfaces st0.0
```

Results

In configuration mode, confirm your interface configuration by entering the `show` command. The configured interfaces are bound to a security zone at the `[edit security]` hierarchy level. Use the `show zones` command to view the zones, where the configured interfaces (gr-, st0.0, and lo0) are displayed. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Parameters for configuring the GRE interfaces:

```
user@host> show interfaces
ge-0/0/0 {
    unit 0 {
        family inet {
            address 10.10.11.1/24;
        }
    }
}

gr-0/0/0 {
    unit 0 {
        tunnel {
            source 172.20.1.1;
            destination 172.20.1.2;
        }
        family inet {
            address 192.168.1.1/24;
        }
    }
}

lo0 {
    unit 0 {
        family inet {
            address 172.20.1.1/32;
        }
    }
}

st0 {
    unit 0 {
        family inet;
    }
}

[edit]
root@Juniper# show
routing-options {
    static {
        route 10.10.10.0/24 next-hop gr-0/0/0.0;
```

```
        route 172.20.1.2/32 next-hop st0.0;
    }
}
```

Parameters for configuring the GRE interfaces with security zones:

```
[edit security]
root@Juniper# show
zones {
    security-zone trust {
        host-inbound-traffic {
            system-services {
                all;
            }
            protocols {
                all;
            }
        }
        interfaces {
            gr-0/0/0.0;
            lo0.0;
            st0.0;
        }
    }
}
```

Verification

IN THIS SECTION

- Verification of the IPsec Tunnel | 294

Verification of the IPsec Tunnel

Purpose

Verify that the IPsec tunnel is up.

Action

Run the commands `show security ike security-associations` and `show security ipsec security-associations` commands.

SEE ALSO

Generic Routing Encapsulation (GRE)

Understanding Generic Routing Encapsulation

Verifying That Generic Routing Encapsulation Tunneling Is Working Correctly

Example: Configuring a GRE Tunnel When the Tunnel Destination Is in a Routing Instance

IN THIS SECTION

Requirements

Overview

You can configure a GRE tunnel when the tunnel destination is in a default routing instance or non-default routing instance. Configuration of a GRE tunnel requires defining the tunnel source and the tunnel destination addresses. If the tunnel destination is in a routing instance, and there is more than one routing instance present, you need to specify the correct routing instance and also the routing table to be used to reach the configured tunnel destination address.

NOTE: The tunnel destination address is by default considered to be reachable using the default routing table "inet.0".

Configuration

IN THIS SECTION

In this example, you can configure a GRE tunnel between the gr- interfaces on SRX Series Firewalls with two instances. The instances are when the tunnel destination is in a default routing instance and when the tunnel destination is in a non-default routing instance.

Configuring a GRE Tunnel When the Tunnel Destination Is in a Default Routing Instance

This example uses the default routing instance to reach the tunnel destination. Because of this, the routing table inet.0 is used by default.

Step-by-Step Procedure

1. Specify the source and destination address of the tunnel.

```
[edit interfaces]
user@host# set interfaces gr-0/0/0  unit 0 tunnel source 172.16.0.1 destination 10.10.1.2
user@host# set interfaces gr-0/0/0  unit 0 family inet 192.168.100.1/30;
```

2. Configure the ge- interface and lo0 interface with the family set as inet.

```
[edit interfaces]
user@host# set interfaces ge-0/0/0 unit 0 family inet address 172.30.73.56/24
user@host# set interfaces lo0 unit 0 family inet address 172.16.0.1/32
```

3. Configure the GRE tunnel interface for routing options as mentioned in the GRE configuration topic.

Configuring a GRE Tunnel When the Tunnel Destination Is in a Non-default Routing Instance

For a non-default routing instance, ensure that you have already configured the gr-0/0/0 interface.

Step-by-Step Procedure

1. Configure the GRE tunnel with the gr-0/00 interface and family set as inet.

```
[edit interfaces]
user@host# set interfaces gr-0/00 unit 0 family inet address
```

2. Specify the source and destination address of the tunnel.

```
[edit interfaces]
user@host# set interfaces gr-0/0/0  unit 0 tunnel source 172.16.0.1 tunnel destination
10.10.1.2 family inet 192.168.100.1/30;
```

3. Configure the ge- interface and lo0 interface with the family set as inet.

```
[edit interfaces]
user@host# set interfaces ge-0/0/0 unit 0 family inet address 172.30.73.56/24
user@host# set interfaces lo0 unit 0 family inet address 172.16.0.1/32
```

4. Configure the routing instances used for the tunnel interface.

```
[edit routing-instances]
user@host# set routing-instances test instance-type virtual-router
user@host# set routing-instances test routing-options static route 10.10.1.2/32 next-hop
172.30.73.57
user@host# set routing-instances test interface ge-0/0/0.0
```

5. Configure the routing-instance for GRE tunnel interfaces.

```
[edit interfaces]
user@host# set interfaces gr-0/0/0 unit 0  tunnel routing-instance destination test
```

6. Add the static route for tunnel destination.

```
[edit interfaces]
user@host# set routing-options static route 10.10.1.2/32 next-table test.inet.0
```

NOTE: When the SRX Series Firewall is in packet mode, you do not need to configure a static route to make the tunnel destination reachable from inet.0. However, you still need to specify the correct routing instance under the gr-0/0/0 interface.

Results

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

When the tunnel destination is in a default routing instance:

```
interfaces {
    gr-0/0/0 {
        unit 0 {
            tunnel {
                source 172.16.0.1;
                destination 10.10.1.2;
            }
            family inet {
                address 192.168.100.1/30;
            }
        }
    }
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 172.30.73.56/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 172.16.0.1/32;
            }
        }
    }
    ...
}
routing-options {
```

```
    static {
        route 10.10.1.2/32 next-hop 172.30.73.57;                    # Tunnel destination is
reachable from default routing-instance
        ...
    }
}
routing-instances {
    test {
        instance-type virtual-router;
        interface gr-0/0/0.0;
        routing-options {
            ...
        }
    }
}
```

When the tunnel destination is in a non-default routing instance:

```
interfaces {
    gr-0/0/0 {
        unit 0 {
            tunnel {
                source 172.16.0.1;
                destination 10.10.1.2;
                routing-instance {
                    destination test;                                # Routing-instance to reach
tunnel destination
                }
            }
            family inet {
                address 192.168.100.1/30;
            }
        }
    }
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 172.30.73.56/24;
            }
        }
    }
    lo0 {
```

```
        unit 0 {
            family inet {
                address 172.16.0.1/32;
            }
        }
    }
    ...
}

routing-options {
         static {
         route 10.10.1.2/32 next-table test.inet.0;                # Tunnel destination is
reachable via test.inet.0
     ...
     }
}

routing-instances {
     test {
         instance-type virtual-router;
         interface ge-0/0/0;
         routing-options {
             static {
                 route 10.10.1.2/32 next-hop 172.30.73.57;         # Tunnel destination is
reachable from non-default routing-instance
                 ...
             }
         }
     }
}
```

Verification

IN THIS SECTION

Verification of Static Route Use

Purpose

Verify that the static route is used.

Action

Run the `show route forwarding table` command.

```
user@host> show route forwarding-table table test
No Title
Routing table: test.inet
Internet:
Enabled protocols: Bridging,
Destination        Type RtRef Next hop           Type Index    NhRef Netif
default            perm     0                    rjct      597     1
0.0.0.0/32         perm     0                    dscd      590     1
10.10.1.2/32       user     1 172.30.73.57       hold      598     4 ge-0/0/0.0
172.16.0.1.10.10.1.2.47/72
                   dest     0                    locl      617     1
172.30.73.0/24     intf     0                    rslv      588     1 ge-0/0/0.0
172.30.73.0/32     dest     0 172.30.73.0        recv      586     1 ge-0/0/0.0
172.30.73.56/32    intf     0 172.30.73.56       locl      587     2
172.30.73.56/32    dest     0 172.30.73.56       locl      587     2
172.30.73.57/32    dest     0 172.30.73.57       hold      598     4 ge-0/0/0.0
172.30.73.255/32   dest     0 172.30.73.255      bcst      585     1 ge-0/0/0.0
224.0.0.0/4        perm     0                    mdsc      596     1
224.0.0.1/32       perm     0 224.0.0.1          mcst      600     1
255.255.255.255/32 perm     0                    bcst      601     1
```

Verification of Static Route Used in Default Instance

Purpose

Verify that the static route is used for the default instance.

Action

Run the `show route forwarding table` command.

```
user@host> show route forwarding-table matching 10.10.1.2
Routing table: default.inet
Internet:
Enabled protocols: Bridging,
Destination        Type RtRef Next hop           Type Index    NhRef Netif
10.10.1.2/32       user     0                    rtbl      604     3
```

SEE ALSO

Generic Routing Encapsulation (GRE)

Understanding Generic Routing Encapsulation

Verifying That Generic Routing Encapsulation Tunneling Is Working Correctly

RELATED DOCUMENTATION

Generic Routing Encapsulation (GRE)

Configuring Point-to-Point Protocol over Ethernet

IN THIS SECTION

Point-to-Point Protocol over Ethernet (PPPoE) combines PPP, with the Ethernet link-layer protocol that allows users to connect to a network of hosts over a bridge or access concentrator. The below topics discuss the overview of PPPoE interfaces, PPPoE Ethernet interfaces, CHAP aunthentication on PPPoE, displaying statistics, setting tracing options for PPPoE and verification of these interfaces on security devices.

Understanding Point-to-Point Protocol over Ethernet

IN THIS SECTION

Point-to-Point Protocol over Ethernet (PPPoE) combines PPP, which typically runs over broadband connections, with the Ethernet link-layer protocol that allows users to connect to a network of hosts over a bridge or access concentrator. PPPoE enables service providers to maintain access control through PPP connections and also manage multiple hosts at a remote site.

PPPoE connects multiple hosts on an Ethernet LAN to a remote site through a single customer premises equipment (CPE) device—a Juniper Networks device. Hosts share a common digital subscriber line (DSL), a cable modem, or a wireless connection to the Internet.

To use PPPoE, you must initiate a PPPoE session, encapsulate Point-to-Point Protocol (PPP) packets over Ethernet, and configure the device as a PPPoE client. To provide a PPPoE connection, each PPP session must learn the Ethernet address of the remote peer and establish a unique session identifier during the PPPoE discovery and session stages.

PPPoE has two stages, the discovery stage and the PPPoE session stage. In the discovery stage, the client discovers the access concentrator by identifying the Ethernet media access control (MAC) address of the access concentrator and establishing a PPPoE session ID. In the session stage, the client and the access concentrator build a point-to-point connection over Ethernet, based on the information collected in the discovery stage.

This topic contains the following sections:

PPPoE Discovery Stage

To initiate a PPPoE session, a host must first identify the Ethernet MAC address of the remote peer and establish a unique PPPoE session ID for the session. Learning the remote Ethernet MAC address is called PPPoE discovery.

During the PPPoE discovery process, the host does not discover a remote endpoint on the Ethernet network. Instead, the host discovers the access concentrator through which all PPPoE sessions are established. Discovery is a client/server relationship, with the host (a device running Junos OS) acting as the client and the access concentrator acting as the server. Because the network might have more than one access concentrator, the discovery stage allows the client to communicate with all of them and select one.

NOTE: A device cannot receive PPPoE packets from two different access concentrators on the same physical interface.

The PPPoE discovery stage consists of the following steps:

1. PPPoE Active Discovery Initiation (PADI)—The client initiates a session by broadcasting a PADI packet to the LAN to request a service.
2. PPPoE Active Discovery Offer (PADO)—Any access concentrator that can provide the service requested by the client in the PADI packet replies with a PADO packet that contains its own name, the unicast address of the client, and the service requested. An access concentrator can also use the PADO packet to offer other services to the client.
3. PPPoE Active Discovery Request (PADR)—From the PADOs it receives, the client selects one access concentrator based on its name or the services offered and sends it a PADR packet to indicate the service or services needed.
4. PPPoE Active Discovery Session-Confirmation (PADS)—When the selected access concentrator receives the PADR packet, it accepts or rejects the PPPoE session:
    - To accept the session, the access concentrator sends the client a PADS packet with a unique session ID for a PPPoE session and a service name that identifies the service under which it accepts the session.

- To reject the session, the access concentrator sends the client a PADS packet with a service name error and resets the session ID to zero.

PPPoE Session Stage

The PPPoE session stage starts after the PPPoE discovery stage is over. The access concentrator can start the PPPoE session after it sends a PADS packet to the client, or the client can start the PPPoE session after it receives a PADS packet from the access concentrator. A device supports multiple PPPoE sessions on each interface, but no more than 256 PPPoE sessions per device.

Each PPPoE session is uniquely identified by the Ethernet address of the peer and the session ID. After the PPPoE session is established, data is sent as in any other PPP encapsulation. The PPPoE information is encapsulated within an Ethernet frame and is sent to a unicast address. Magic numbers, echo requests, and all other PPP traffic behave exactly as in normal PPP sessions. In this stage, both the client and the server must allocate resources for the PPPoE *logical interface*.

After a session is established, the client or the access concentrator can send a PPPoE Active Discovery Termination (PADT) packet anytime to terminate the session. The PADT packet contains the destination address of the peer and the session ID of the session to be terminated. After this packet is sent, the session is closed to PPPoE traffic.

NOTE: If PPPoE session is already up and the user restarts the PPPoE daemon, a new PPPoE daemon with a new PID starts while the existing session is not terminated.

If PPPoE session is already down and user restarts the PPPoE daemon, the PPPoE discovery establishes a new session.

The PPPoE session is not terminated for the following configuration changes:

- Changing idle time out value
- Changing auto rec timer value
- Deleting idle time out
- Deleting auto rec timer
- Add new auto rec time
- Add new idle time out
- Change negotiate address to static address
- Change static ip address to a new static ip address
- Changing default chap secrete

The PPPoE session is terminated for the following configuration changes:

- Add ac name
- Delete chap ppp options
- Add new chap ppp options
- Configure uifd mac

> **NOTE**: When the MTU for an underlying physical interface is changed, it brings down the PPPoE session. The PPPoE MTU can be greater than 1492 if the Ethernet or WAN connection supports RFC 4638 (Mini Jumbo Frames).

SEE ALSO

Understanding Physical Encapsulation on an Interface | 269

Understanding PPPoE Interfaces | 306

Understanding PPPoE Ethernet Interfaces | 316

Understanding the PPPoE-Based Radio-to-Router Protocol

Understanding PPPoE Interfaces

The device's Point-to-Point Protocol over Ethernet (PPPoE) interface to the access concentrator can be a Fast Ethernet interface, a Gigabit Ethernet interface, or a redundant Ethernet interface. The PPPoE configuration is the same for all interfaces. The only difference is the encapsulation for the underlying interface to the access concentrator:

- If the interface is Ethernet, use a PPPoE encapsulation.

To configure a PPPoE interface, you create an interface with a *logical interface* unit 0, then specify a logical Ethernet interface as the underlying interface for the PPPoE session. You then specify other PPPoE options, including the access concentrator and PPPoE session parameters.

> **NOTE**: PPPoE over redundant Ethernet (reth) interface is supported on SRX100, SRX210, SRX220, SRX240, SRX300, SRX320, SRX340 and SRX650 devices. (Platform support depends on the Junos OS release in your installation.) This feature allows an

existing PPPoE session to continue without starting a new PPP0E session in the event of a failover.

Example: Configuring PPPoE Interfaces

IN THIS SECTION

This example shows how to configure a PPPoE interface.

Requirements

Before you begin, configure an Ethernet interface. See "Example: Creating an Ethernet Interface" on page 130.

Overview

In this example, you create the PPPoE interface pp0.0 and specify the logical Ethernet interface ge-0/0/1.0 as the underlying interface. You also set the access concentrator, set the PPPoE session parameters, and set the MTU of the IPv4 family to 1492.

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

```
set interfaces pp0 unit 0 pppoe-options underlying-interface ge-0/0/1.0 access-concentrator
ispl.com auto-reconnect 100 idle-timeout 100 client service-name video@ispl.com
set interfaces pp0 unit 0  family inet mtu 1492 negotiate-address
```

Step-by-Step Procedure

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

To configure a PPPoE interface:

1. Create a PPPoE interface.

```
[edit]
user@host# edit interfaces pp0 unit 0
```

2. Configure PPPoE options.

```
[edit interfaces pp0 unit 0]
user@host# set pppoe-options underlying-interface ge-0/0/1.0 access-concentrator ispl.com
auto-reconnect 100 idle-timeout 100 client service-name video@ispl.com
```

3. Configure the MTU.

```
[edit interfaces pp0 unit 0]
user@host# set family inet mtu 1492
```

NOTE: If you want to configure `mtu` to a value above 1492 octets, then use `ppp-max-payload` option. Refer *pppoe-options* for more details.

4. Configure the PPPoE interface address.

```
[edit interfaces pp0 unit 0]
user@host# set family inet negotiate-address
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces pp0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interfaces pp0
unit 0 {
    pppoe-options {
        underlying-interface ge-0/0/1.0;
        idle-timeout 100;
        access-concentrator ispl.com;
        service-name "vide0@ispl.com";
        auto-reconnect 100;
        client;
     }
    family inet {
        mtu 1492;
        negotiate-address;
        }
    }
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying PPPoE Interfaces

Purpose

Verify that the PPPoE device interfaces are configured properly.

Action

From operational mode, enter the `show interfaces pp0` command.

```
user@host> show interfaces pp0
Physical interface: pp0, Enabled, Physical link is Up
  Interface index: 67, SNMP ifIndex: 317
  Type: PPPoE, Link-level type: PPPoE, MTU: 9192
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Link type      : Full-Duplex
  Link flags     : None
  Last flapped   : Never
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface pp0.0 (Index 1) (SNMP ifIndex 330)
    Flags: Point-To-Point SNMP-Traps 16384 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 3304,
      Session AC name: isp1.com, AC MAC address: 00:90:1a:40:f6:4c,
      Service name: video@isp1.com, Configured AC name: isp1.com,
      Auto-reconnect timeout: 60 seconds
      Underlying interface: ge-5/0/0.0 (Index 71)
  Input packets : 23
  Output packets: 22
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 16 (00:00:26 ago), Output: 0 (never)
```

```
  LCP state: Opened
  NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls:
  Not-configured
  CHAP state: Success
    Protocol inet, MTU: 1492
      Flags: Negotiate-Address
      Addresses, Flags: Kernel Is-Preferred Is-Primary
        Destination: 211.211.211.2, Local: 211.211.211.1
```

The output shows information about the physical and the logical interfaces. Verify the following information:

- The physical interface is enabled and the link is up.
- The PPPoE session is running on the correct logical interface.
- For state, the state is active (up).
- For underlying interface, the physical interface on which the PPPoE session is running is correct:
  - For an Ethernet connection, the underlying interface is Fast Ethernet or Gigabit Ethernet—for example, ge-5/0/0.0.

Verifying PPPoE Sessions

Purpose

Verify that a PPPoE session is running properly on the logical interface.

Action

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

```
user@host> show pppoe interfaces
pp0.0 Index 67
  State: Session up, Session ID: 31,
  Service name: video@isp1.com, Configured AC name: isp1.com,
  Session AC name: belur, AC MAC address: 00:90:1a:40:f6:4e,
  Auto-reconnect timeout: 1 seconds,
  Underlying interface: ge-0/0/1.0 Index 69
```

The output shows information about the PPPoE sessions. Verify the following information:

- The PPPoE session is running on the correct logical interface.

- For state, the session is active (up).
- For underlying interface, the physical interface on which the PPPoE session is running is correct:
  - For an Ethernet connection, the underlying interface is Fast Ethernet or Gigabit Ethernet—for example, ge-0/0/1.0.

NOTE: To clear a PPPoE session on the pp0.0 interface, use the `clear pppoe sessions pp0.0` command. To clear all sessions on the interface, use the `clear pppoe sessions` command.

Verifying the PPPoE Version

Purpose

Verify the version information of the PPPoE protocol configured on the device interfaces.

Action

From operational mode, enter the `show pppoe version` command.

```
user@host> show pppoe version
Point-to-Point Protocol Over Ethernet, version 1. rfc2516
   PPPoE protocol               = Enabled
   Maximum Sessions             = 256
   PADI resend timeout          = 2 seconds
   PADR resend timeout          = 16 seconds
   Max resend timeout           = 64 seconds
   Max Configured AC timeout    = 4 seconds
```

The output shows PPPoE protocol information. Verify the following information:

- The correct version of the PPPoE protocol is configured on the interface.
- For PPPoE protocol, the PPPoE protocol is enabled.

Verifying PPPoE Statistics

Purpose

Verify the statistics information about PPPoE interfaces.

Action

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

```
user@host> show pppoe statistics
Active PPPoE sessions: 4
  PacketType                       Sent         Received
    PADI                            502                0
    PADO                              0              219
    PADR                            219                0
    PADS                              0              219
    PADT                              0              161
    Service name error                0                0
    AC system error                   0               13
    Generic error                     0                0
    Malformed packets                 0               41
    Unknown packets                   0                0
  Timeout
    PADI                             42
    PADO                              0
    PADR                              0
```

The output shows information about active sessions on PPPoE interfaces. Verify the following information:

- Total number of active PPPoE sessions running on the interfac
- For packet type, the number of packets of each type sent and received during the PPPoE session

Disabling the End-of-List Tag

IN THIS SECTION

During the PPPoE discovery stage, any access concentrator that can provide the service requested by the client in the PADI packet replies with a PADO packet that contains its own name, the unicast address of the client, and the service requested. An access concentrator can also use the PADO packet

to offer other services to the client. When a client receives a PADO packet, and if it encounters the `End-of-List` tag in the PADO packet, tags after the `End-of-List` tag are ignored and the complete information is not processed correctly. As a result, the PPPoE connection is not established correctly.

Starting in Junos OS Release 12.3X48-D10 you can avoid some PPPoE connection errors by configuring the `ignore-eol-tag` option to disable the `End-of-List` tag in the PADO packet.

Procedure

Step-by-Step Procedure

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

To disable the `End-of-List` tag:

1. Create a PPPoE interface.

```
[edit]
user@host# set interfaces pp0 unit 0
```

2. Configure PPPoE options.

```
[edit interfaces pp0 unit 0]
user@host# set pppoe-options ignore-eol-tag
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces pp0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show interfaces pp0
unit 0 {
pppoe-options {
    ignore-eol-tag;
 }
```

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

Verifying That the End-of-List Tag Is Disabled

Purpose

Verify the status of the `End-of-List` tag in the PPPoE configuration.

Action

From operational mode, enter the `show interfaces pp0.0` command.

```
user@host> show pppoe interfaces pp0.0
Logical interface pp0.0 (Index 78) (SNMP ifIndex 541)
    Flags: Point-To-Point SNMP-Traps 0x0 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 3,
      Session AC name: cell, Remote MAC address: 00:26:88:f7:77:83,
      Configured AC name: None, Service name: None,
      Auto-reconnect timeout: Never, Idle timeout: Never,
      Underlying interface: ge-0/0/3.0 (Index 77)
     Ignore End-Of-List tag: Enable
```

```
user@host> show pppoe interfaces pp0.0 extensive
pp0.0 Index 74
  State: Session up, Session ID: 1,
  Service name: None,
  Session AC name: cell, Configured AC name: None,
  Remote MAC address: 00:26:88:f7:77:83,
  Session uptime: 00:02:03 ago,
  Auto-reconnect timeout: 10 seconds, Idle timeout: Never,
  Underlying interface: ge-0/0/3.0 Index 73
  Ignore End-of-List tag: Enable
  PacketType                      Sent         Received
    PADI                            23                0
    PADO                             0                5
    PADR                            11                0
    PADS                             0                2
    PADT                             2                0
    Service name error               0                0
    AC system error                  0                0
    Generic error                    0                0
```

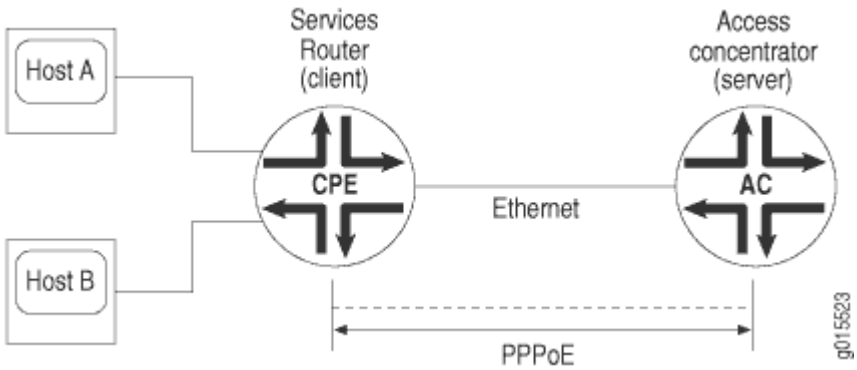
```
    Malformed packets                0                0
    Unknown packets                  0                0
  Timeout
    PADI                             3
    PADO                             0
    PADR                             3
  Receive Error Counters
    PADI                             0
    PADO                             0
    PADR                             0
    PADS                             0
```

The output shows information about active sessions on PPPoE interfaces. Verify that the `Ignore End-of-List tag: Enable` option is set.

Understanding PPPoE Ethernet Interfaces

During a Point-to-Point Protocol over Ethernet (PPPoE) session, the device encapsulates each PPP frame in an Ethernet frame and transports the frames over an Ethernet loop. Figure 16 on page 316 shows a typical PPPoE session between a device and an access concentrator on the Ethernet loop.

Figure 16: PPPoE Session on the Ethernet Loop



To configure PPPoE on an Ethernet interface, you configure encapsulation on the *logical interface*.

Example: Configuring PPPoE Encapsulation on an Ethernet Interface

IN THIS SECTION

This example shows how to configure PPPoE encapsulation on an Ethernet interface.

Requirements

Before you begin:

- Configure an Ethernet interface. See "Example: Creating an Ethernet Interface" on page 130.
- Configure a PPPoE encapsulation interface. See "Example: Configuring PPPoE Interfaces" on page 307.

Overview

In this example, you configure PPPoE encapsulation on the ge-0/0/1 interface.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To configure PPPoE encapsulation:

1. Enable PPPoE encapsulation on the interface.

```
[edit]
user@host# set interfaces ge-0/0/1 unit 0 encapsulation ppp-over-ether
```

2. Commit the configuration if you are done configuring the device.

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

Verification

To verify the configuration is working properly, enter the `show interfaces ge-0/0/1` command.

Example: Configuring PPPoE Encapsulation on an ATM-over-ADSL Interface

IN THIS SECTION

This example shows how to configure a physical interface for Ethernet over ATM encapsulation and how to create a logical interface for PPPoE over LLC encapsulation.

Requirements

Before you begin:

- Configure network interfaces. See "Example: Creating an Ethernet Interface" on page 130.
- Configure PPPoE interfaces. See "Example: Configuring PPPoE Interfaces" on page 307.

- Configure PPPoE encapsulation on an Ethernet interface. See "Example: Configuring PPPoE Encapsulation on an Ethernet Interface" on page 317.

Overview

In this example, you configure the physical interface at-2/0/0 for Ethernet over ATM encapsulation. As part of the configuration, you set the virtual path identifier (VPI) on an ATM-over-ADSL physical interface to 0, you set the ADSL operating mode to auto, and you set the encapsulation type to ATM-over-ADSL. Then you create a logical interface for PPPoE over LLC encapsulation.

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

```
set interfaces at-2/0/0 atm-options vpi 0
set interfaces at-2/0/0 dsl-options operating-mode auto
set interfaces at-2/0/0 encapsulation ethernet-over-atm
set interfaces at-2/0/0 unit 0 encapsulation ppp-over-ether-over-atm-llc vci 0.120
```

Step-by-Step Procedure

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

To configure PPPoE encapsulation on an ATM-over-ADSL interface:

1. Configure the physical interface.

```
[edit]
user@host# edit interfaces at-2/0/0
```

2. Set the VPI on the interface.

```
[edit interfaces at-2/0/0]
user@host# set atm-options vpi 0
```

3. Configure the ADSL operating mode.

```
[edit interfaces at-2/0/0]
user@host# set dsl-options operating-mode auto
```

4. Configure PPPoE encapsulation.

```
[edit interfaces at-2/0/0]
user@host# set encapsulation ethernet-over-atm
```

5. Create a logical interface and configure LLC encapsulation.

```
[edit interfaces at-2/0/0]
user@host# set unit 0 encapsulation ppp-over-ether-over-atm-llc vci 0.120
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces at-2/0/0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show interfaces at-2/0/0 {
    encapsulation ethernet-over-atm;
        atm-options {
        vpi 0;
    }
```

```
    dsl-options {
        operating-mode auto;
     }
    unit 0 {
        encapsulation ppp-over-ether-over-atm-llc;
        vci 0.120;
        }
    }
```

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

Verification

IN THIS SECTION

- Verifying a PPPoE Configuration for an ATM-over-ADSL or ATM-over-SHDSL Interface | 321

Confirm that the configuration is working properly.

Verifying a PPPoE Configuration for an ATM-over-ADSL or ATM-over-SHDSL Interface

Purpose

Verify the PPPoE configuration for an ATM-over-ADSL or ATM-over-SHDSL interface.

Action

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

Understanding CHAP Authentication on a PPPoE Interface

For interfaces with Point-to-Point Protocol over Ethernet (PPPoE) encapsulation, you can configure interfaces to support the PPP Challenge Handshake Authentication Protocol (CHAP). When you enable CHAP on an interface, the interface can authenticate its peer and be authenticated by its peer.

If you set the `passive` option to handle incoming CHAP packets only, the interface does not challenge its peer. However, if the interface is challenged, it responds to the challenge. If you do not set the `passive` option, the interface always challenges its peer.

You can configure Remote Authentication Dial-In User Service (RADIUS) authentication of PPP sessions using CHAP. CHAP enables you to send RADIUS messages through a routing instance to customer RADIUS servers in a private network.

Example: Configuring CHAP Authentication on a PPPoE Interface

IN THIS SECTION

This example shows how to configure CHAP authentication on a PPPoE interface.

Requirements

Before you begin:

- Configure an Ethernet interface. See "Example: Creating an Ethernet Interface" on page 130.
- Configure a PPPoE interface. See "Example: Configuring PPPoE Interfaces" on page 307.

Overview

In this example, you configure a CHAP access profile, and then apply it to the PPPoE interface pp0. You also configure the hostname to be used in CHAP challenge and response packets, and set the passive option for handling incoming CHAP packets.

Configuration

IN THIS SECTION

- Procedure | 323

Procedure

CLI Quick Configuration

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

```
set access profile A-ppp-client client client1 chap-secret my-secret
set interfaces pp0 unit 0 ppp-options chap access-profile A-ppp-client local-name A-ge-0/0/1.0
passive
```

Step-by-Step Procedure

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

To configure CHAP on a PPPoE interface:

1. Configure a CHAP access profile.

```
[edit]
user@host# set access profile A-ppp-client client client1 chap-secret my-secret
```

2. Enable CHAP options on the interface.

```
[edit]
user@host# edit interfaces pp0 unit 0 ppp-options chap
```

3. Configure the CHAP access profile on the interface.

```
[edit interfaces pp0 unit 0 ppp-options chap]
user@host# set access-profile A-ppp-client
```

4. Configure a hostname for the CHAP challenge and response packets.

```
[edit interfaces pp0 unit 0 ppp-options chap]
user@host# set local-name A-ge-0/0/1.0
```

5. Set the passive option to handle incoming CHAP packets only.

```
[edit interfaces pp0 unit 0 ppp-options chap]
user@host# set passive
```

Results

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

```
[edit]
    user@host# show interfaces
    pp0 {
    unit 0 {
     ppp-options {
    chap {
    access-profile A-ppp-client;
    local-name A-ge-0/0/1.0;
    passive;
    }
    }
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying CHAP Authentication

Purpose

Verify that CHAP is enabled on the interface.

Action

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

Verifying Credit-Flow Control

IN THIS SECTION

Purpose

Display PPPoE credit-flow control information about credits on each side of the PPPoE session when credit processing is enabled on the interface.

Action

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

```
pp0.51 Index 73
  State: Session up, Session ID: 3,
  Service name: None,
  Configured AC name: None, Session AC name: None,
  Remote MAC address: 00:22:83:84:2e:81,
  Session uptime: 00:05:48 ago,
  Auto-reconnect timeout: Never, Idle timeout: Never,
  Underlying interface: ge-0/0/4.1 Index 72
  PADG Credits: Local: 12345, Remote: 6789, Scale factor: 128 bytes
  PADQ Current bandwidth: 750 Kbps, Maximum 1000 Kbps
    Quality: 85, Resources 65, Latency 100 msec.
  Dynamic bandwidth: 3 Kbps
pp0.1000 Index 71
  State: Down, Session ID: 1,
  Service name: None,
  Configured AC name: None, Session AC name: None,
  Remote MAC address: 00:00:00:00:00:00,
  Auto-reconnect timeout: Never, Idle timeout: Never,
  Underlying interface: ge-0/0/1.0 Index 70
  PADG Credits: enabled
  Dynamic bandwidth: enabled
```

Verifying PPPoE Interfaces

IN THIS SECTION

Purpose

Display PPPoE interfaces information.

Action

- To display PPPoE interface information:

```
user@host> show pppoe interfaces pp0.51 detail
```

```
pp0.51 Index 75
  State: Session up, Session ID: 1,
  Service name: None,
  Configured AC name: None, Session AC name: None,
  Remote MAC address: 00:11:22:33:44:55,
  Session uptime: 00:04:18 ago,
  Auto-reconnect timeout: Never, Idle timeout: Never,
  Underlying interface: ge-0/0/1.0 Index 70
  PADQ Current bandwidth: 750 Kbps, Maximum 1000 Kbps
    Quality: 85, Resources 65, Latency 100 msec.
  Dynamic bandwidth: 3 Kbps
```

- To display PPPoE terse interface information:

```
user@host> show pppoe interfaces terse pp0.51
```

```
Interface    Admin Link Proto    Local        Remote
          pp0.51       up  up   inet   5.1.1.1     --> 5.1.1.2
                                  inet6    fe80::21f:12ff:fed2:2918/64
                                 feee::5:1:1:1/126
```

Verifying R2CP Interfaces

IN THIS SECTION

Purpose

Display R2CP interfaces information.

Action

- To display R2CP interface information:

```
root@host> show r2cp interfaces
```

```
Interface: ge-0/0/3.51
  Nodes: 0
```

- To display R2CP information:

```
root@host> show r2cp radio extensive
```

```
    Node Packet Type          Sent       Received         Errors
    MIM                          -              1              0
    ROM                          1              -              -
    Heartbeats                   0              0              0
    Node Term                    0              0              0
    Node Term Ack                0              0              -

    Heartbeat Timeouts              0
    Node Term Timeouts              0
```

```
    Session Packet Type        Sent        Received          Errors
    Init                          -               1               0
    Init ACK                      1               -               -
    Update                        -               0               0
    Terminate                     0               0               0
    Terminate ACK                 0               0               0

    Terminate Timeouts               0
```

- To display R2CP session information:

```
root@host> show r2cp sessions extensive
```

```
Session: 1
  Destination MAC address 01:02:03:04:05:06
   Status: Established VLANs 201
   Virtual channel:  2
   Session Update:  last received: 3.268 seconds
     Current bandwidth: 22000 Kbps, Maximum 22000 Kbps
     Quality: 100, Resources 100, Latency 100 msec.
   Effective bandwidth: 952 Kbps, last change: 51.484 seconds
     Updates below threshold: 1


    Session Packet Type        Sent        Received          Errors
    Init                          -               1               0
    Init ACK                      1               -               -
    Update                        -               0               0
    Terminate                     0               0               0
    Terminate ACK                 0               0               0

    Terminate Timeouts               0
```

Displaying Statistics for PPPoE

IN THIS SECTION

Purpose

Display PPPoE statistics.

Action

```
user@host> show interfaces pp0.51 statistics
```

```
Logical interface pp0.51 (Index 75) (SNMP ifIndex 137)
    Flags: Point-To-Point SNMP-Traps 0x0 Encapsulation: PPPoE
    PPPoE:
      State: SessionUp, Session ID: 1,
      Session AC name: None, Remote MAC address: 00:22:83:84:2f:03,
      Underlying interface: ge-0/0/4.1 (Index 74)
    Input packets : 20865
    Output packets: 284636
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 0 (never), Output: 943 (00:00:06 ago)
  LCP state: Opened
  NCP state: inet: Opened, inet6: Opened, iso: Not-configured, mpls:
  Not-configured
  CHAP state: Closed
  PAP state: Closed
    Security: Zone: Null
    Protocol inet, MTU: 1492
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 5.1.1.2, Local: 5.1.1.1
    Protocol inet6, MTU: 1492
```

```
      Flags: None
      Addresses, Flags: Is-Preferred
        Destination: fe80::/64, Local: fe80::21f:12ff:fed2:2918
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: feee::5:1:1:0/126, Local: feee::5:1:1:1
```

Setting Tracing Options for PPPoE

To trace the operations of the router's PPPoE process, include the traceoptions statement at the [edit protocols pppoe] hierarchy level:

```
[edit protocols pppoe]
traceoptions {
    file filename <files number> <match regular-expression> <size size> <world-readable | no-world-readable>;
    flag flag;
    level severity-level;
    no-remote-trace;
}
```

To specify more than one tracing operation, include multiple `flag` statements.

You can specify the following flags in the `traceoptions` statement:

- `all`—All areas of code
- `config`—Configuration code
- `events`—Event code
- `gres`—Gres code
- `init`—Initialization code
- `interface-db`—Interface database code
- `memory`—Memory management code
- `protocol`—PPPoE protocol processing code
- `rtsock`—Routing socket code
- `session-db`—Session management code

- signal—Signal handling code
- state—State handling code
- timer—Timer code
- ui—User interface code

Change History Table

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

Release	Description
12.3X48-D10	Starting in Junos OS Release 12.3X48-D10 you can avoid some PPPoE connection errors by configuring the ignore-eol-tag option to disable the End-of-List tag in the PADO packet.

Configure the PPPoE Family for an Underlying Interface

SUMMARY

Describes the PPPoE encapsulation and its limitations. This topic also describes the introduction of "family PPPoE" that can support other protocol families like IPv4 and IPv6 to coexist on the underlying interface.

Use PPPoE encapsulation on the physical interface to dedicate the interface to PPPoE. PPPoE does not support other protocol families on that interface. Configuring IPv4 with PPPoE and IPv6 on the same interface can create protocol conflicts and operational complexities. To address these issues, configure a new PPPoE family such as IPv4 or IPv6. This approach simplifies operations and resolves conflicts.

You can have the PPPoE family configuration on ge, xe, and reth interfaces only. Also, you cannot configure an interface with both “encapsulation ppp-over-ether” and “family pppoe”.

Benefits

The PPPoE configuration supports IPv4 traffic over PPPoE and IPv6 traffic over Ethernet without PPPoE on the same interface. This configuration eliminates separate interfaces, simplifying network architecture and management.

Configuration

To configure the PPPoE family over an underlying interface, use the following command:

```
[edit interfaces ge-0/0/1 unit logical-unit-number]
user@host# set family pppoe
```

Sample Configuration

When you already have ppp-over-ethernet encapsulation configuration, you can't just replace with family pppoe.

For example, the following configuration shows a PPPoE with encapsulation:

```
set interfaces ge-0/0/11 unit 0 encapsulation ppp-over-ether
set interfaces pp0 unit 0 ppp-options pap local-name SRX2
set interfaces pp0 unit 0 ppp-options pap local-password "$9$WhBLX-bwgJZjVb.PQz6/"
set interfaces pp0 unit 0 ppp-options pap passive
set interfaces pp0 unit 0 pppoe-options underlying-interface ge-0/0/11.0
set interfaces pp0 unit 0 pppoe-options idle-timeout 0
set interfaces pp0 unit 0 pppoe-options auto-reconnect 10
set interfaces pp0 unit 0 pppoe-options client
set interfaces pp0 unit 0 family inet mtu 1492
set interfaces pp0 unit 0 family inet negotiate-address
```

To add PPPoE family configuration, you need to:

1. `delete interface ge-0/0/11`
2. `delete interface pp0`
3. `commit`

Add the PPPoE family configuration once the commit is complete. Note that you need to add pp0 configuration as well.

```
set interfaces ge-0/0/11 unit 0 family pppoe
set interfaces pp0 unit 0 ppp-options pap local-name SRX2
set interfaces pp0 unit 0 ppp-options pap local-password "$9$WhBLX-bwgJZjVb.PQz6/"
set interfaces pp0 unit 0 ppp-options pap passive
```

```
set interfaces pp0 unit 0 pppoe-options underlying-interface ge-0/0/11.0
set interfaces pp0 unit 0 pppoe-options idle-timeout 0
set interfaces pp0 unit 0 pppoe-options auto-reconnect 10
set interfaces pp0 unit 0 pppoe-options client
set interfaces pp0 unit 0 family inet mtu 1492
set interfaces pp0 unit 0 family inet negotiate-address
```

5
CHAPTER

Configuring Link Services Interfaces

IN THIS CHAPTER

Configuring Link Services Interfaces

IN THIS SECTION

Juniper Networks devices support link services on the `lsq-0/0/0` link services queuing interface which includes multilink services like MLPP, MLFR and CRTP. The topics below discuss the overview of link services, configuration details and verification of the link services.

Link Services Interfaces Overview

IN THIS SECTION

Link services include the multilink services Multilink Point-to-Point Protocol (MLPPP), Multilink Frame Relay (MLFR), and Compressed Real-Time Transport Protocol (CRTP). Juniper Networks devices support link services on the `lsq-0/0/0` link services queuing interface.

You configure the link services queuing interface (`lsq-0/0/0`) on a Juniper Networks device to support multilink services and CRTP.

The link services queuing interface consists of services provided by the following interfaces: multilink services interface (`ml-`*`fpc/pic/port`*), link services interface (`ls-`*`fpc/pic/port`*), and link services intelligent queuing interface (`lsq-`*`fpc/pic/port`*). Although the multilink services, link services, and link services intelligent queuing (IQ) interfaces are installed on Physical Interface Cards (PICs), the link services queuing interface is an internal interface only and is not associated with a physical medium or *Physical Interface Module* (PIM).

NOTE: (`ls-`*`fpc/pic/port`*) is not supported.

This section contains the following topics.

Services Available on a Link Services Interface

The link services interface is a *logical interface* available by default. Table 47 on page 337 summarizes the services available on the interface.

Table 47: Services Available on a Link Services Interface

Services	Purpose	More Information
Multilink bundles by means of MLPPP and MLFR encapsulation	Aggregates multiple constituent links into one larger logical bundle to provide additional bandwidth, load balancing, and redundancy. **NOTE**: Dynamic call admission control (DCAC) configurations are not supported on Link Services Interfaces.	• "Example: Configuring an MLPPP Bundle" on page 388
Link fragmentation and interleaving (LFI)	Reduces delay and *jitter* on links by breaking up large data packets and interleaving delay-sensitive voice packets with the resulting smaller packets.	"Understanding Link Fragmentation and Interleaving Configuration" on page 369

Table 47: Services Available on a Link Services Interface *(Continued)*

Services	Purpose	More Information
Compressed Real-Time Transport Protocol (CRTP)	Reduces the overhead caused by Real-Time Transport Protocol (RTP) on voice and video packets.	"Compressed Real-Time Transport Protocol Overview" on page 341
Class-of-service (CoS) classifiers, forwarding classes, schedulers and scheduler maps, and shaping rates	Provides a higher priority to delay-sensitive packets—by configuring CoS, such as the following: • Classifiers—To classify different types of traffic, such as voice, data, and network control packets. • Forwarding classes—To direct different types of traffic to different output queues. • Fragmentation map—To define mapping between forwarding class and multilink class, and forwarding class and fragment threshold. In forwarding class and multilink class mapping, drop timeout can be configured. • Schedulers and scheduler maps—To define properties for the output queues such as delay-buffer, transmission rate, and transmission priority. • Shaping rate—To define certain bandwidth usage by an interface.	• "Example: Configuring Interface Shaping Rates" on page 385 • "Configuring Fragmentation by Forwarding Class" on page 342

Link Services Exceptions

The link and multilink services implementation is similar to the implementation on other routing platforms, with the following exceptions:

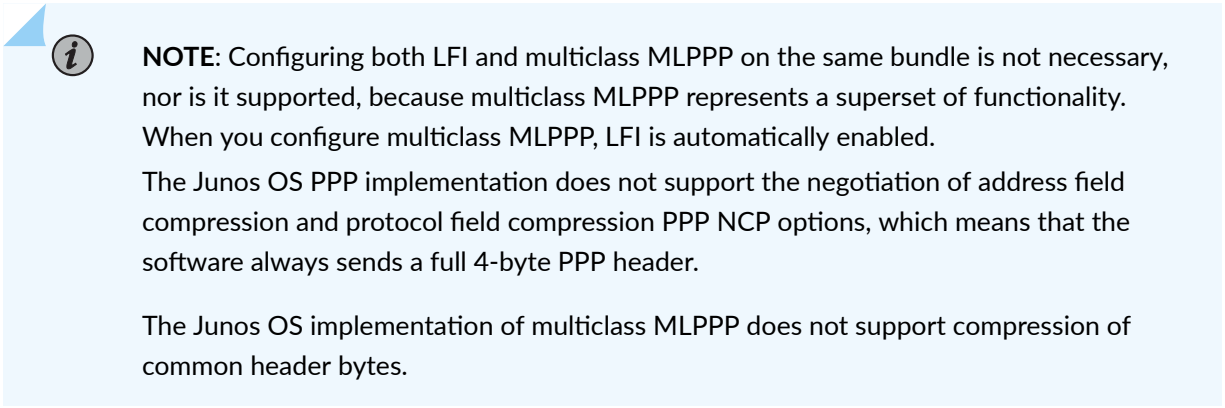
- Support for link and multilink services are on the `lsq-0/0/0` interface instead of the `ml-`*`fpc/pic/port`* , `lsq-`*`fpc/pic/port`* , and `ls-`*`fpc/pic/port`* interfaces.

- When LFI is enabled, fragmented packets are queued in a round-robin fashion on the constituent links to enable per-packet and per-fragment load balancing. See "Queuing with LFI" on page 340.
- Support for per-unit scheduling is on all types of constituent links (on all types of interfaces).
- Support for Compressed Real-Time Transport Protocol (CRTP) is for both MLPPP and PPP.

Configuring Multiclass MLPPP

For `lsq-0/0/0` on Juniper Networks device, with MLPPP encapsulation, you can configure multiclass MLPPP. If you do not configure multiclass MLPPP, fragments from different classes cannot be interleaved. All fragments for a single packet must be sent before the fragments from another packet are sent. Non-fragmented packets can be interleaved between fragments of another packet to reduce latency seen by non-fragmented packets. In effect, latency-sensitive traffic is encapsulated as regular PPP traffic, and bulk traffic is encapsulated as multilink traffic. This model works as long as there is a single class of latency-sensitive traffic, and there is no high-priority traffic that takes precedence over latency-sensitive traffic. This approach to LFI, used on the Link Services PIC, supports only two levels of traffic priority, which is not sufficient to carry the four-to-eight forwarding classes.

Multiclass MLPPP makes it possible to have multiple classes of latency-sensitive traffic that are carried over a single multilink bundle with bulk traffic. In effect, multiclass MLPPP allows different classes of traffic to have different latency guarantees. With multiclass MLPPP, you can map each forwarding class into a separate multilink class, thus preserving priority and latency guarantees.



NOTE: Configuring both LFI and multiclass MLPPP on the same bundle is not necessary, nor is it supported, because multiclass MLPPP represents a superset of functionality. When you configure multiclass MLPPP, LFI is automatically enabled.

The Junos OS PPP implementation does not support the negotiation of address field compression and protocol field compression PPP NCP options, which means that the software always sends a full 4-byte PPP header.

The Junos OS implementation of multiclass MLPPP does not support compression of common header bytes.

Multiclass MLPPP greatly simplifies packet ordering issues that occur when multiple links are used. Without multiclass MLPPP, all voice traffic belonging to a single flow is hashed to a single link to avoid packet ordering issues. With multiclass MLPPP, you can assign voice traffic to a high-priority class, and you can use multiple links.

To configure multiclass MLPPP on a link services IQ interface, you must specify how many multilink classes should be negotiated when a link joins the bundle, and you must specify the mapping of a forwarding class into an multiclass MLPPP class.

To specify how many multilink classes should be negotiated when a link joins the bundle, include the `multilink-max-classes` statement:

```
multilink-max-classes number;
```

You can include this statement at the following hierarchy levels:

- `[edit interfaces interface-name unit logical-unit-number]`
- `[edit logical-routers logical-router-name interfaces interface-name unit logical-unit-number]`

The number of multilink classes can be 1 through 8. The number of multilink classes for each forwarding class must not exceed the number of multilink classes to be negotiated.

To specify the mapping of a forwarding class into a multiclass MLPPP class, include the `multilink-class` statement at the `[edit class-of-service fragmentation-maps forwarding-class class-name]` hierarchy level:

```
edit class-of-service fragmentation-maps forwarding-class
class-name
multilink-class number
```

The multilink class index number can be 0 through 7. The `multilink-class` statement and the `no-fragmentation` statement are mutually exclusive.

To view the number of multilink classes negotiated, issue the `show interfaces lsq-0/0/0.logical-unit-number detail` command.

Queuing with LFI

LFI or non-LFI packets are placed into queues on constituent links based on the queues in which they arrive. No changes in the queue number occur while the fragmented, non-fragmented, or LFI packets are being queued.

For example, assume that Queue Q0 is configured with fragmentation threshold 128, Q1 is configured with no fragmentation, and Q2 is configured with fragmentation threshold 512. Q0 is receiving stream of traffic with packet size 512. Q1 is receiving voice traffic of 64 bytes, and Q2 is receiving stream of traffic with 128-byte packets. Next the stream on Q0 gets fragmented and queued up into Q0 of a constituent link. Also, all packets on Q2 are queued up on Q0 on constituent link. The stream on Q1 is considered to be LFI because no fragmentation is configured. All the packets from Q0 and Q2 are queued up on Q0 of constituent link. All the packets from Q1 are queued up on Q2 of constituent link.

Using `lsq-0/0/0`, CRTP can be applied on LFI and non-LFI packets. There will be no changes in their queue numbers because of CRTP.

Queuing on Q2s of Constituent Links

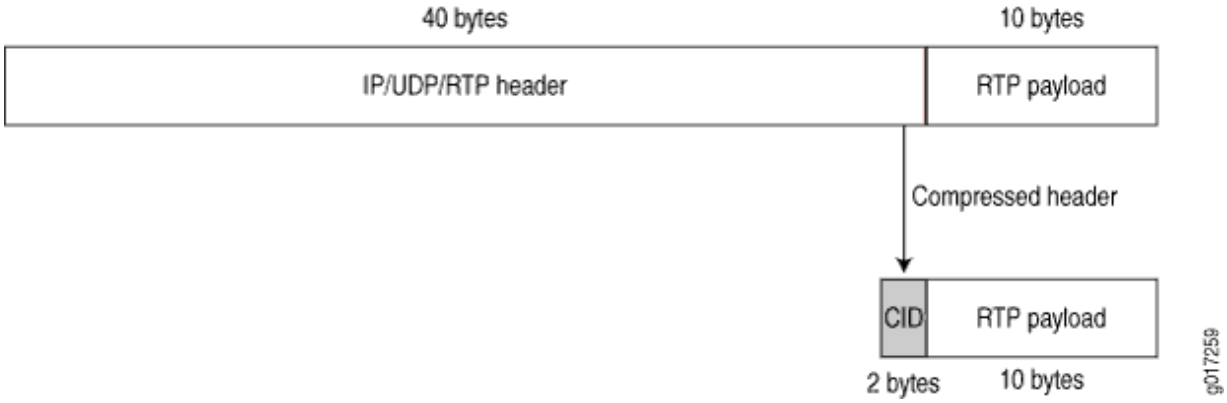
When using *class of service* on a multilink bundle, all Q2 traffic from the multilink bundle is queued to Q2 of constituent links based on a hash computed from the source address, destination address, and the IP protocol of the packet. If the IP payload is TCP or UDP traffic, the hash also includes the source port and destination port. As a result of this hash algorithm, all traffic belonging to one traffic flow is queued to Q2 of one constituent link. This method of traffic delivery to the constituent link is applied at all times, including when the bundle has not been set up with LFI.

Compressed Real-Time Transport Protocol Overview

Real-Time Transport Protocol (RTP) can help achieve interoperability among different implementations of network audio and video applications. However, in some cases, the header, which includes the IP, UDP, and RTP headers, can be too large (around 40 bytes) on networks using low-speed lines such as dial-up modems. Compressed Real-Time Transport Protocol (CRTP) can be configured to reduce network overhead on low-speed links. CRTP replaces the IP, UDP, and RTP headers with a 2-byte context ID (CID), reducing the header overhead considerably.

Figure 17 on page 341 shows how CRTP compresses the RTP header in a voice packet by reducing a 40-byte header to a 2-byte header.

Figure 17: CRTP



You can configure CRTP with MLPPP or PPP logical interface encapsulation on link services interfaces. See "Example: Configuring an MLPPP Bundle" on page 388.

Real-time and non-real-time data frames are carried together on lower-speed links without causing excessive delays to the real-time traffic. See "Understanding Link Fragmentation and Interleaving Configuration" on page 369.

Configuring Fragmentation by Forwarding Class

For `lsq-0/0/0`, you can specify fragmentation properties for specific forwarding classes. Traffic on each forwarding class can be either multilink encapsulated (fragmented and sequenced) or non-encapsulated (hashed with no fragmentation). By default, traffic in all forwarding classes is multilink encapsulated.

When you do not configure fragmentation properties for the queues on MLPPP interfaces, the fragmentation threshold you set at the `[edit interfaces` *`interface-name`* `unit` *`logical-unit-number`* `fragment-threshold]` hierarchy level is the fragmentation threshold for all forwarding classes within the MLPPP interface. For MLFR FRF.16 interfaces, the fragmentation threshold you set at the `[edit interfaces` *`interface-name`* `mlfr-uni-nni-bundle-options fragment-threshold]` hierarchy level is the fragmentation threshold for all forwarding classes within the MLFR FRF.16 interface.

If you do not set a maximum fragment size anywhere in the configuration, packets are still fragmented if they exceed the smallest maximum transmission unit (MTU) or maximum received reconstructed unit (MRRU) of all the links in the bundle. A non-encapsulated flow uses only one link. If the flow exceeds a single link, then the forwarding class must be multilink encapsulated, unless the packet size exceeds the MTU/MRRU.

Even if you do not set a maximum fragment size anywhere in the configuration, you can configure the MRRU by including the mrru statement at the `[edit interfaces` *`lsq-0/0/0`* `unit` *`logical-unit-number`*`]` or `[edit interfaces` *`interface-name`* `mlfr-uni-nni-bundle-options]` hierarchy level. The MRRU is similar to the MTU, but is specific to link services interfaces. By default the MRRU size is 1504 bytes, and you can configure it to be from 1500 through 4500 bytes.

To configure fragmentation properties on a queue, include the fragmentation-maps statement at the `[edit class-of-service]` hierarchy level:

```
[edit class-of-service]
```

```
fragmentation-maps {
    map-name {
        forwarding-class class-name {
            fragment-threshold bytes;
            multilink-class number;
            no-fragmentation;
        }
```

```
    }
}
```

To set a per-forwarding class fragmentation threshold, include the `fragment-threshold` statement in the fragmentation map. This statement sets the maximum size of each multilink fragment.

To set traffic on a queue to be non-encapsulated rather than multilink encapsulated, include the `no-fragmentation` statement in the fragmentation map. This statement specifies that an extra fragmentation header is not prepended to the packets received on this queue and that static link load balancing is used to ensure in-order packet delivery.

For a given forwarding class, you can include either the `fragment-threshold` or `no-fragmentation` statement; they are mutually exclusive.

You use the `multilink-class` statement to map a forwarding class into a multiclass MLPPP. For a given forwarding class, you can include either the `multilink-class` or `no-fragmentation` statement; they are mutually exclusive.

To associate a fragmentation map with a multilink PPP interface or MLFR FRF.16 DLCI, include the `fragmentation-map` statement at the `[edit class-of-service interfaces` *`interface-name`* `unit` **logical-unit-number**`]` hierarchy level:

```
[edit class-of-service interfaces]
```

```
lsq-0/0/0 {
    unit logical-unit-number { # Multilink PPP
    fragmentation-map map-name;
    }
}
```

```
lsq-0/0/0:channel { # MLFR FRF.16
    unit logical-unit-number
    fragmentation-map map-name;
    }
}
```

Configuring Link-Layer Overhead

Link-layer overhead can cause packet drops on constituent links because of bit stuffing on serial links. Bit stuffing is used to prevent data from being interpreted as control information.

By default, 4 percent of the total bundle bandwidth is set aside for link-layer overhead. In most network environments, the average link-layer overhead is 1.6 percent. Therefore, we recommend 4 percent as a safeguard.

For `lsq-0/0/0` on Juniper Networks device, you can configure the percentage of bundle bandwidth to be set aside for link-layer overhead. To do this, include the link-layer-overhead statement:

```
link-layer-overhead percent;
```

You can include this statement at the following hierarchy levels:

- `[edit interfaces interface-name mlfr-uni-nni-bundle-options]`
- `[edit interfaces interface-name unit logical-unit-number]`
- `[edit logical-routers logical-router-name interfaces interface-name unit logical-unit-number]`

You can configure the value to be from 0 percent through 50 percent.

Link Services Configuration Overview

Before you begin:

- Install device hardware.
- Establish basic connectivity. See the Getting Started Guide for your device.
- Have a basic understanding of physical and logical interfaces and Juniper Networks interface conventions. See "Understanding Interfaces" on page 2

Plan how you are going to use the link services interface on your network. See "Link Services Interfaces Overview" on page 336.

To configure link services on an interface, perform the following tasks:

1. Configure link fragmentation and interleaving (LFI). See "Example: Configuring Link Fragmentation and Interleaving" on page 370.
2. Configure classifiers and forwarding classes. See "Example: Defining Classifiers and Forwarding Classes" on page 373.
3. Configure scheduler maps. See "Understanding How to Define and Apply Scheduler Maps" on page 378.
4. Configure interface shaping rates. See "Example: Configuring Interface Shaping Rates" on page 385
5. Configure an MLPPP bundle. See "Example: Configuring an MLPPP Bundle" on page 388.

6. To configure CRTP, see "Example: Configuring the Compressed Real-Time Transport Protocol" on page 394

Verifying the Link Services Interface

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying Link Services Interface Statistics

IN THIS SECTION

Purpose

Verify the link services interface statistics.

Action

The sample output provided in this section is based on the configurations provided in "Example: Configuring an MLPPP Bundle" on page 388. To verify that the constituent links are added to the bundle correctly and the packets are fragmented and transmitted correctly, take the following actions:

1. On device R0 and device R1, the two devices used in this example, configure MLPPP and LFI as described in "Example: Configuring an MLPPP Bundle" on page 388.
2. From the CLI, enter the `ping` command to verify that a connection is established between R0 and R1.
3. Transmit 10 data packets, 200 bytes each, from R0 to R1.

4. On R0, from the CLI, enter the `show interfaces` *interface-name* `statistics` command.

```
user@R0> show interfaces lsq-0/0/0 statistics detail
Physical interface: lsq-0/0/0, Enabled, Physical link is Up
  Interface index: 134, SNMP ifIndex: 29, Generation: 135
  Link-level type: LinkService, MTU: 1504
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Last flapped   : 2006-06-23 11:36:23 PDT (03:38:43 ago)
  Statistics last cleared: 2006-06-23 15:13:12 PDT (00:01:54 ago)
  Traffic statistics:
   Input  bytes  :                    0                    0 bps
   Output bytes  :                 1820                    0 bps
   Input  packets:                    0                    0 pps
   Output packets:                   10                    0 pps
  ...
  Egress queues: 8 supported, 8 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 DATA                          10                   10                    0
    1 expedited-fo                   0                    0                    0
    2 VOICE                          0                    0                    0
    3 NC                             0                    0                    0

  Logical interface lsq-0/0/0.0 (Index 67) (SNMP ifIndex 41) (Generation 133)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: Multilink-PPP
    Bandwidth: 16mbps
    Bundle options:
      ...
      Drop timer period              0
      Sequence number format         long (24 bits)
      Fragmentation threshold        128
      Links needed to sustain bundle  1
      Interleave fragments           Enabled
    Bundle errors:
      Packet drops                   0 (0 bytes)
      Fragment drops                 0 (0 bytes)
     ...
    Statistics         Frames        fps         Bytes          bps
    Bundle:
      Fragments:
        Input :             0          0             0            0
        Output:            20          0          1920            0
```

```
      Packets:
        Input :              0          0             0            0
        Output:             10          0          1820            0
    Link:
      se-1/0/0.0
        Input :              0          0             0            0
        Output:             10          0          1320            0
      se-1/0/1.0
        Input :              0          0             0            0
        Output:             10          0           600            0
...
      Destination: 10.0.0.9/24, Local: 10.0.0.10, Broadcast: Unspecified, Generation:144
```

This output shows a summary of interface information. Verify the following information:

- `Physical interface`—The physical interface is `Enabled`. If the interface is shown as `Disabled`, do either of the following:
  - In the CLI configuration editor, delete the `disable` statement at the `[edit interfaces` *`interface-name`*`]` level of the configuration hierarchy.
  - In the J-Web configuration editor, clear the `Disable` check box on the `Interfaces>`*`interface-name`* page.
- `Physical link`—The physical link is `Up`. A link state of `Down` indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- `Last flapped`—The `Last Flapped` time is an expected value. The `Last Flapped` time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates likely link-layer errors.
- `Traffic statistics`—Number and rate of bytes and packets received and transmitted on the interface. Verify that the number of inbound and outbound bytes and packets match the expected throughput for the physical interface. To clear the statistics and see only new changes, use the `clear interfaces statistics` *`interface-name`* command.
- `Queue counters`—Name and number of queues are as configured. This sample output shows that 10 data packets were transmitted and no packets were dropped.
- `Logical interface`—Name of the multilink bundle you configured—`lsq-0/0/0.0`.
- `Bundle options`—Fragmentation threshold is correctly configured, and fragment interleaving is enabled.
- `Bundle errors`—Any packets and fragments dropped by the bundle.

- `Statistics`—The fragments and packets are received and transmitted correctly by the device. All references to traffic direction (input or output) are defined with respect to the device. Input fragments received by the device are assembled into input packets. Output packets are segmented into output fragments for transmission out of the device.

  In this example, 10 data packets of 200 bytes were transmitted. Because the fragmentation threshold is set to 128 bytes, all data packets were fragmented into two fragments. The sample output shows that 10 packets and 20 fragments were transmitted correctly.

- `Link`—The constituent links are added to this bundle and are receiving and transmitting fragments and packets correctly. The combined number of fragments transmitted on the constituent links must be equal to the number of fragments transmitted from the bundle. This sample output shows that the bundle transmitted 20 fragments and the two constituent links `se-1/0/0.0` and `se-1/0/1.0.0` correctly transmitted `10+10=20` fragments.

- `Destination` and `Local`—IP address of the remote side of the multilink bundle and the local side of the multilink bundle. This sample output shows that the destination address is the address on R1 and the local address is the address on R0.

Verifying Link Services CoS Configuration

IN THIS SECTION

Purpose

Verify CoS configurations on the link services interface.

Action

From the CLI, enter the following commands:

- `show class-of-service interface` *`interface-name`*
- `show class-of-service classifier name` *`classifier-name`*
- `show class-of-service scheduler-map` *`scheduler-map-name`*

The sample output provided in this section is based on the configurations provided in"Example: Configuring an MLPPP Bundle" on page 388.

```
user@R0> show class-of-service interface lsq-0/0/0
Physical interface: lsq-0/0/0, Index: 136
Queues supported: 8, Queues in use: 4
  Scheduler map: [default], Index: 2
  Input scheduler map: [default], Index: 3
  Chassis scheduler map: [default-chassis], Index: 4
 Logical interface: lsq-0/0/0.0, Index: 69
    Object              Name                    Type              Index
    Scheduler-map       s_map                   Output            16206
    Classifier          ipprec-compatibility    ip                   12
```

```
user@R0> show class-of-service interface ge-0/0/1
Physical interface: ge-0/0/1, Index: 140
          Queues supported: 8, Queues in use: 4
          Scheduler map: [default], Index: 2
          Input scheduler map: [default], Index: 3

         Logical interface: ge-0/0/1.0, Index: 68
         Object              Name                  Type         Index
         Classifier         classfy_input           ip           4330
```

```
user@R0> show class-of-service classifier name classify_input
Classifier: classfy_input, Code point type: inet-precedence, Index: 4330

  Code point       Forwarding class           Loss priority
    000                 DATA                      low
    010                 VOICE                     low
```

```
user@R0> show class-of-service scheduler-map s_map
Scheduler map: s_map, Index: 16206

  Scheduler: DATA, Forwarding class: DATA, Index: 3810
  Transmit rate: 49 percent, Rate Limit: none, Buffer size: 49 percent, Priority:low
    Drop profiles:
      Loss priority        Protocol         Index         Name
```

```
    Low                    any             1         [default-drop-profile]
    Medium low             any             1         [default-drop-profile]
    Medium high            any             1         [default-drop-profile]
    High                   any             1         [default-drop-profile]

Scheduler: VOICE, Forwarding class: VOICE, Index: 43363
Transmit rate: 50 percent, Rate Limit: none, Buffer size: 5 percent, Priority:high
  Drop profiles:
    Loss priority        Protocol          Index        Name
    Low                     any              1        [default-drop-profile]
    Medium low              any              1        [default-drop-profile]
    Medium high             any              1        [default-drop-profile]
    High                    any              1        [default-drop-profile]

Scheduler: NC, Forwarding class: NC, Index: 2435
Transmit rate: 1 percent, Rate Limit: none, Buffer size: 1 percent, Priority:high
  Drop profiles:
    Loss priority        Protocol         Index      Name
    Low                     any              1       [default-drop-profile]
    Medium low              any              1       [default-drop-profile]
    Medium high             any              1       [default-drop-profile]
    High                    any              1       [default-drop-profile]
```

These output examples show a summary of configured CoS components. Verify the following information:

- `Logical Interface`—Name of the multilink bundle and the CoS components applied to the bundle. The sample output shows that the multilink bundle is `lsq-0/0/0.0`, and the CoS scheduler-map `s_map` is applied to it.
- `Classifier`—Code points, forwarding classes, and loss priorities assigned to the classifier. The sample output shows that a default classifier, `ipprec-compatibility`, was applied to the `lsq-0/0/0` interface and the classifier `classify_input` was applied to the `ge-0/0/1` interface.
- `Scheduler`—Transmit rate, buffer size, priority, and loss priority assigned to each scheduler. The sample output displays the data, voice, and network control schedulers with all the configured values.

Understanding the Internal Interface LSQ-0/0/0 Configuration

The link services interface is an internal interface only. It is not associated with a physical medium or PIM. Packets are routed to this interface for link bundling or compression.

It may be required that you upgrade your configuration to use the internal interface lsq-0/0/0 as the link services queuing interface instead of ls-0/0/0, which has been deprecated. You can also roll back your modified configuration to use ls-0/0/0.

Example: Upgrading from ls-0/0/0 to lsq-0/0/0 for Multilink Services

IN THIS SECTION

This example shows how to upgrade from ls-0/0/0 to lsq-0/0/0 (or to reverse the change) for multilink services.

Requirements

This procedure is only necessary if you are still using ls-0/0/0 instead of lsq-/0/0/0 or if you need to revert to the old interface.

Overview

In this example, you rename the link services internal interface from ls-0/0/0 to lsq-0/0/0 or vice versa. You rename all occurrences of ls-0/0/0 in the configuration to lsq-0/0/0 and configure the fragmentation map by adding no fragmentation. You specify no fragmentation after the name of queue 2, if queue 2 is configured, or after assured forwarding. You then attach the fragmentation map configured in the preceding step to lsq-0/0/0 and specify the unit number as 6 of the multilink bundle for which interleave fragments is configured.

Then you roll back the configuration from lsq-0/0/0 to ls-0/0/0. You rename all occurrences in the configuration from lsq-0/0/0 to ls-0/0/0. You delete the fragmentation map if it is configured under the [class-of-service] hierarchy and delete the fragmentation map if it is assigned to lsq-0/0/0. You can delete multilink-max-classes if it is configured for lsq-0/0/0 under the [interfaces] hierarchy. You then delete link-layer-overhead if it is configured for lsq-0/0/0 under the [interfaces] hierarchy.

If no fragmentation is configured on any forwarding class and the fragmentation map is assigned to lsq-0/0/0, then you configure interleave fragments for the ls-0/0/0 interface. Finally, you configure the classifier for LFI packets to refer to queue 2. (The ls-0/0/0 interface treats queue 2 as the LFI queue.)

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

To quickly upgrade from ls-0/0/0 to lsq-0/0/0 (or reverse the change), copy the following commands and paste them into the CLI:

```
For interfaces ls-0/0/0 to lsq-0/0/0
[edit]
rename interfaces ls-0/0/0 to lsq-0/0/0
set class-of-service fragmentation-maps map6 forwarding-class assured-forwarding no-fragmentation
set class-of-service interfaces lsq-0/0/0 unit 6 fragmentation-map map6
```

```
For interfaces lsq-0/0/0 to ls-0/0/0
[edit]
rename interfaces lsq-0/0/0 to ls-0/0/0
delete class-of-service fragmentation-maps map6
delete class-of-service interfaces lsq-0/0/0 unit 6 fragmentation-map map6
delete  interfaces lsq-0/0/0 unit 6 link-layer-overhead
delete  interfaces lsq-0/0/0:0 mlfr-uni-nni-bundle-options link-layer-overhead
set interfaces ls-0/0/0 unit 6 interleave-fragments
```

Step-by-Step Procedure

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

To upgrade from ls-0/0/0 to lsq-0/0/0 or to reverse that change:

1. Rename all the occurrences of ls-0/0/0 in the configuration.

```
[edit]
user@host# rename interfaces ls-0/0/0 to lsq-0/0/0
```

2. Configure the fragmentation map.

```
[edit class-of-service fragmentation-maps]
user@host# set map6 forwarding-class assured-forwarding no-fragmentation
```

3. Specify the unit number of the multilink bundle.

```
[edit class-of-service ]
user@host# set interfaces lsq-0/0/0 unit 6 fragmentation-map map6
```

4. Roll back the configuration for all occurrences in the configuration.

```
[edit]
user@host# rename interfaces lsq-0/0/0 to ls-0/0/0
```

5. Delete fragmentation map under class of service.

```
[edit]
user@host# delete class-of-service fragmentation-maps map6
```

6. Delete fragmentation map if it is assigned to the lsq-0/0/0 interface.

```
[edit class-of-service interfaces]
user@host# delete lsq-0/0/0 unit 6 fragmentation-map map6
```

7. Delete multilink max classes if it is configured for lsq-0/0/0.

NOTE: Multilink-max-classes is not supported and is most likely not configured.

8. Delete link-layer-overhead if it is configured for lsq-0/0/0.

```
[edit interfaces]
user@host# delete lsq-0/0/0 unit 6 link-layer-overhead
```

9. Delete link-layer-overhead if it is configured for lsq-0/0/0:0.

```
[edit interfaces]
user@host# delete lsq-0/0/0:0 mlfr-uni-nni-bundle-options link-layer-overhead
```

10. Configure interleave fragments for the ls-0/0/0 interface.

```
[edit interfaces]
user@host# set ls-0/0/0 unit 6 interleave-fragments
```

Results

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

```
[edit]
    user@host# show class-of-service
    interfaces {
    lsq-0/0/0 {
        unit 6 {
        fragmentation-map map6;
        }
    }
    }
    fragmentation-maps {
        map6 {
        forwarding-class {
        assured-forwarding {
            no-fragmentation;
            }
        }
```

```
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying Link Services Internal Interface ls-0/0/0 to lsq-0/0/0

Purpose

Verify the link services internal interface ls-0/0/0 changed to lsq-0/0/0.

Action

From operational mode, enter the `show class-of-service` command.

Troubleshoot the Link Services Interface

IN THIS SECTION

To solve configuration problems on a link services interface:

Determine Which CoS Components Are Applied to the Constituent Links

IN THIS SECTION

Problem

Description

You are configuring a multilink bundle, but you also have traffic without MLPPP encapsulation passing through constituent links of the multilink bundle. Do you apply all CoS components to the constituent links, or is applying them to the multilink bundle enough?

Solution

You can apply a scheduler map to the multilink bundle and its constituent links. Although you can apply several CoS components with the scheduler map, configure only the ones that are required. We recommend that you keep the configuration on the constituent links simple to avoid unnecessary delay in transmission.

Table 2 shows the CoS components to be applied on a multilink bundle and its constituent links.

Table 48: CoS Components Applied on Multilink Bundles and Constituent Links

Cos Component	Multilink Bundle	Constituent Links	Explanation
Classifier	Yes	No	CoS classification takes place on the incoming side of the interface, not on the transmitting side, so no classifiers are needed on constituent links.

Table 48: CoS Components Applied on Multilink Bundles and Constituent Links *(Continued)*

Cos Component	Multilink Bundle	Constituent Links	Explanation
Forwarding class	Yes	No	Forwarding class is associated with a queue, and the queue is applied to the interface by a scheduler map. The queue assignment is predetermined on the constituent links. All packets from Q2 of the multilink bundle are assigned to Q2 of the constituent link, and packets from all the other queues are queued to Q0 of the constituent link.
Scheduler map	Yes	Yes	Apply scheduler maps on the multilink bundle and the constituent link as follows: • Transmit rate—Make sure that the relative order of the transmit rate configured on Q0 and Q2 is the same on the constituent links as on the multilink bundle. • Scheduler priority—Make sure that the relative order of the scheduler priority configured on Q0 and Q2 is the same on the constituent links as on the multilink bundle. • Buffer size—Because all non-LFI packets from the multilink bundle transit on Q0 of the constituent links, make sure that the buffer size on Q0 of the constituent links is large enough. • RED drop profile—Configure a RED drop profile on the multilink bundle only. Configuring the RED drop profile on the constituent links applies a back pressure mechanism that changes the buffer size and introduces variation. Because this behavior might cause fragment drops on the constituent links, make sure to leave the RED drop profile at the default settings on the constituent links.

Table 48: CoS Components Applied on Multilink Bundles and Constituent Links *(Continued)*

Cos Component	Multilink Bundle	Constituent Links	Explanation
Shaping rate for a per-unit scheduler or an interface-level scheduler	No	Yes	Because per-unit scheduling is applied only at the end point, apply this shaping rate to the constituent links only. Any configuration applied earlier is overwritten by the constituent link configuration.
Transmit-rate exact or queue-level shaping	Yes	No	The interface-level shaping applied on the constituent links overrides any shaping on the queue. Thus apply transmit-rate exact shaping on the multilink bundle only.
Rewrite rules	Yes	No	Rewrite bits are copied from the packet into the fragments automatically during fragmentation. Thus what you configure on the multilink bundle is carried on the fragments to the constituent links.
Virtual channel group	Yes	No	Virtual channel groups are identified through firewall filter rules that are applied on packets only before the multilink bundle. Thus you do not need to apply the virtual channel group configuration to the constituent links.

SEE ALSO

- Class of Service User Guide (Security Devices)

Determine What Causes Jitter and Latency on the Multilink Bundle

IN THIS SECTION

Problem

Description

To test jitter and latency, you send three streams of IP packets. All packets have the same IP precedence settings. After configuring LFI and CRTP, the latency increased even over a noncongested link. How can you reduce jitter and latency?

Solution

To reduce jitter and latency, do the following:

1. Make sure that you have configured a shaping rate on each constituent link.
2. Make sure that you have not configured a shaping rate on the link services interface.
3. Make sure that the configured shaping rate value is equal to the physical interface bandwidth.
4. If shaping rates are configured correctly, and jitter still persists, contact the Juniper Networks Technical Assistance Center (JTAC).

Determine If LFI and Load Balancing Are Working Correctly

IN THIS SECTION

Problem

Description

In this case, you have a single network that supports multiple services. The network transmits data and delay-sensitive voice traffic. After configuring MLPPP and LFI, make sure that voice packets are transmitted across the network with very little delay and jitter. How can you find out if voice packets are being treated as LFI packets and load balancing is performed correctly?

Solution

When LFI is enabled, data (non-LFI) packets are encapsulated with an MLPPP header and fragmented to packets of a specified size. The delay-sensitive, voice (LFI) packets are PPP-encapsulated and interleaved between data packet fragments. Queuing and load balancing are performed differently for LFI and non-LFI packets.

To verify that LFI is performed correctly, determine that packets are fragmented and encapsulated as configured. After you know whether a packet is treated as an LFI packet or a non-LFI packet, you can confirm whether the load balancing is performed correctly.

`Solution Scenario`—Suppose two Juniper Networks devices, R0 and R1, are connected by a multilink bundle `lsq-0/0/0.0` that aggregates two serial links, `se-1/0/0` and `se-1/0/1`. On R0 and R1, MLPPP and LFI are enabled on the link services interface and the fragmentation threshold is set to 128 bytes.

In this example, we used a packet generator to generate voice and data streams. You can use the packet capture feature to capture and analyze the packets on the incoming interface.

The following two data streams were sent on the multilink bundle:

- 100 data packets of 200 bytes (larger than the fragmentation threshold)
- 500 data packets of 60 bytes (smaller than the fragmentation threshold)

The following two voice streams were sent on the multilink bundle:

- 100 voice packets of 200 bytes from source port 100
- 300 voice packets of 200 bytes from source port 200

To confirm that LFI and load balancing are performed correctly:

NOTE: Only the significant portions of command output are displayed and described in this example.

1. Verify packet fragmentation. From operational mode, enter the `show interfaces lsq-0/0/0` command to check that large packets are fragmented correctly.

```
user@R0#> show interfaces lsq-0/0/0
Physical interface: lsq-0/0/0, Enabled, Physical link is Up
  Interface index: 136, SNMP ifIndex: 29
  Link-level type: LinkService, MTU: 1504
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
  Last flapped   : 2006-08-01 10:45:13 PDT (2w0d 06:06 ago)
```

```
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface lsq-0/0/0.0 (Index 69) (SNMP ifIndex 42)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: Multilink-PPP
    Bandwidth: 16mbps
    Statistics         Frames      fps       Bytes        bps
    Bundle:
      Fragments:
        Input :             0        0           0          0
        Output:          1100        0      118800          0
      Packets:
        Input :             0        0           0          0
        Output:          1000        0      112000          0
   ...
    Protocol inet, MTU: 1500
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 9.9.9/24, Local: 9.9.9.10
```

Meaning—The output shows a summary of packets transiting the device on the multilink bundle. Verify the following information on the multilink bundle:

- The total number of transiting packets = 1000
- The total number of transiting fragments=1100
- The number of data packets that were fragmented =100

The total number of packets sent (600 + 400) on the multilink bundle match the number of transiting packets (1000), indicating that no packets were dropped.

The number of transiting fragments exceeds the number of transiting packets by 100, indicating that 100 large data packets were correctly fragmented.

Corrective Action—If the packets are not fragmented correctly, check your fragmentation threshold configuration. Packets smaller than the specified fragmentation threshold are not fragmented.

2. Verify packet encapsulation. To find out whether a packet is treated as an LFI or non-LFI packet, determine its encapsulation type. LFI packets are PPP encapsulated, and non-LFI packets are encapsulated with both PPP and MLPPP. PPP and MLPPP encapsulations have different overheads resulting in different-sized packets. You can compare packet sizes to determine the encapsulation type.

A small unfragmented data packet contains a PPP header and a single MLPPP header. In a large fragmented data packet, the first fragment contains a PPP header and an MLPPP header, but the consecutive fragments contain only an MLPPP header.

PPP and MLPPP encapsulations add the following number of bytes to a packet:

- PPP encapsulation adds 7 bytes:

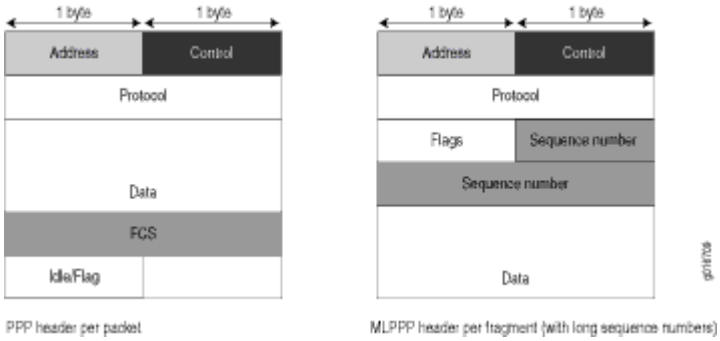
  4 bytes of header+2 bytes of frame check sequence (FCS)+1 byte that is idle or contains a flag

- MLPPP encapsulation adds between 6 and 8 bytes:

  4 bytes of PPP header+2 to 4 bytes of multilink header

Figure 2 shows the overhead added to PPP and MLPPP headers.

Figure 18: PPP and MLPPP Headers



For CRTP packets, the encapsulation overhead and packet size are even smaller than for an LFI packet. For more information, see Example: Configuring the Compressed Real-Time Transport Protocol.

Table 3 shows the encapsulation overhead for a data packet and a voice packet of 70 bytes each. After encapsulation, the size of the data packet is larger than the size of the voice packet.

Table 49: PPP and MLPPP Encapsulation Overhead

Packet Type	Encapsulation	Initial Packet Size	Encapsulation Overhead	Packet Size after Encapsulation
Voice packet (LFI)	PPP	70 bytes	4 + 2 + 1 = 7 bytes	77 bytes

Table 49: PPP and MLPPP Encapsulation Overhead *(Continued)*

Packet Type	Encapsulation	Initial Packet Size	Encapsulation Overhead	Packet Size after Encapsulation
Data fragment (non-LFI) with short sequence	MLPPP	70 bytes	4 + 2 + 1 + 4 + 2 = 13 bytes	83 bytes
Data fragment (non-LFI) with long sequence	MLPPP	70 bytes	4 + 2 + 1 + 4 + 4 = 15 bytes	85 bytes

From operational mode, enter the `show interfaces queue` command to display the size of transmitted packet on each queue. Divide the number of bytes transmitted by the number of packets to obtain the size of the packets and determine the encapsulation type.

3. Verify load balancing. From operational mode, enter the `show interfaces queue` command on the multilink bundle and its constituent links to confirm whether load balancing is performed accordingly on the packets.

```
user@R0> show interfaces queue lsq-0/0/0
Physical interface: lsq-0/0/0, Enabled, Physical link is Up
  Interface index: 136, SNMP ifIndex: 29
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :               600                 0 pps
    Bytes                :             44800                 0 bps
  Transmitted:
    Packets              :               600                 0 pps
    Bytes                :             44800                 0 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
    …
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                 0                 0 pps
    Bytes                :                 0                 0 bps
```

```
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :                400                 0 pps
    Bytes                :              61344                 0 bps
  Transmitted:
    Packets              :                400                 0 pps
    Bytes                :              61344                 0 bps
    …
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                  0                 0 pps
    Bytes                :                  0                 0 bps
 …
```

```
user@R0> show interfaces queue se-1/0/0
Physical interface: se-1/0/0, Enabled, Physical link is Up
  Interface index: 141, SNMP ifIndex: 35
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                350                 0 pps
    Bytes                :              24350                 0 bps
  Transmitted:
    Packets              :                350                 0 pps
    Bytes                :              24350                 0 bps
   ...
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                  0                 0 pps
    Bytes                :                  0                 0 bps
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :                100                 0 pps
    Bytes                :              15272                 0 bps
  Transmitted:
    Packets              :                100                 0 pps
    Bytes                :              15272                 0 bps
    …
```

```
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                19                0 pps
    Bytes                :               247                0 bps
  Transmitted:
    Packets              :                19                0 pps
    Bytes                :               247                0 bps
    …
```

```
user@R0> show interfaces queue se-1/0/1
Physical interface: se-1/0/1, Enabled, Physical link is Up
  Interface index: 142, SNMP ifIndex: 38
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :               350                0 pps
    Bytes                :             24350                0 bps
  Transmitted:
    Packets              :               350                0 pps
    Bytes                :             24350                0 bps
    …
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                 0                0 pps
    Bytes                :                 0                0 bps
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :               300                0 pps
    Bytes                :             45672                0 bps
  Transmitted:
    Packets              :               300                0 pps
    Bytes                :             45672                0 bps
    …
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                18                0 pps
    Bytes                :               234                0 bps
  Transmitted:
```

```
    Packets              :              18              0 pps
    Bytes                :             234              0 bps
```

Meaning—The output from these commands shows the packets transmitted and queued on each queue of the link services interface and its constituent links. Table 4 shows a summary of these values. (Because the number of transmitted packets equaled the number of queued packets on all the links, this table shows only the queued packets.)

Table 50: Number of Packets Transmitted on a Queue

Packets Queued	Bundle lsq-0/0/0.0	Constituent Link se-1/0/0	Constituent Link se-1/0/1	Explanation
Packets on Q0	600	350	350	The total number of packets transiting the constituent links (350+350 = 700) exceeded the number of packets queued (600) on the multilink bundle.
Packets on Q2	400	100	300	The total number of packets transiting the constituent links equaled the number of packets on the bundle.
Packets on Q3	0	19	18	The packets transiting Q3 of the constituent links are for keepalive messages exchanged between constituent links. Thus no packets were counted on Q3 of the bundle.

On the multilink bundle, verify the following:

- The number of packets queued matches the number transmitted. If the numbers match, no packets were dropped. If more packets were queued than were transmitted, packets were dropped because the buffer was too small. The buffer size on the constituent links controls congestion at the output stage. To correct this problem, increase the buffer size on the constituent links.
- The number of packets transiting Q0 (600) matches the number of large and small data packets received (100+500) on the multilink bundle. If the numbers match, all data packets correctly transited Q0.

- The number of packets transiting Q2 on the multilink bundle (400) matches the number of voice packets received on the multilink bundle. If the numbers match, all voice LFI packets correctly transited Q2.

On the constituent links, verify the following:

- The total number of packets transiting Q0 (350+350) matches the number of data packets and data fragments (500+200). If the numbers match, all the data packets after fragmentation correctly transited Q0 of the constituent links.

  Packets transited both constituent links, indicating that load balancing was correctly performed on non-LFI packets.

- The total number of packets transiting Q2 (300+100) on constituent links matches the number of voice packets received (400) on the multilink bundle. If the numbers match, all voice LFI packets correctly transited Q2.

  LFI packets from source port `100` transited `se-1/0/0`, and LFI packets from source port `200` transited `se-1/0/1`. Thus all LFI (Q2) packets were hashed based on the source port and correctly transited both constituent links.

`Corrective Action`—If the packets transited only one link, take the following steps to resolve the problem:

a. Determine whether the physical link is `up` (operational) or `down` (unavailable). An unavailable link indicates a problem with the PIM, interface port, or physical connection (link-layer errors). If the link is operational, move to the next step.

b. Verify that the classifiers are correctly defined for non-LFI packets. Make sure that non-LFI packets are not configured to be queued to Q2. All packets queued to Q2 are treated as LFI packets.

c. Verify that at least one of the following values is different in the LFI packets: source address, destination address, IP protocol, source port, or destination port. If the same values are configured for all LFI packets, the packets are all hashed to the same flow and transit the same link.

4. Use the results to verify load balancing.

Determine Why Packets Are Dropped on a PVC Between a Juniper Networks Device and a Third-Party Device

IN THIS SECTION

Problem

Description

You are configuring a permanent virtual circuit (PVC) between T1, E1, T3, or E3 interfaces on a Juniper Networks device and a third-party device, and packets are being dropped and ping fails.

Solution

If the third-party device does not have the same FRF.12 support as the Juniper Networks device or supports FRF.12 in a different way, the Juniper Networks device interface on the PVC might discard a fragmented packet containing FRF.12 headers and count it as a "Policed Discard."

As a workaround, configure multilink bundles on both peers, and configure fragmentation thresholds on the multilink bundles.

Configuring Link Fragmentation and Interleaving

IN THIS SECTION

The factor that determines the order in which output interface transmits traffic from an output queue is the priority scheduling on a multilink bundle. The large packets using this multilink bundle, cause delay

for the small and delay-sensitive packets to reach their turn for transmission. This delay renders some slow links useless for delay-sensitive traffic. Link fragmentation and interleaving (LFI) solves this problem. The topics below topics the LFI in detail and its configuration.

Understanding Link Fragmentation and Interleaving Configuration

As it does on any other interface, priority scheduling on a multilink bundle determines the order in which an output interface transmits traffic from an output queue. The queues are serviced in a weighted round-robin fashion. But when a queue containing large packets starts using the multilink bundle, small and delay-sensitive packets must wait their turn for transmission. Because of this delay, some slow links can become useless for delay-sensitive traffic.

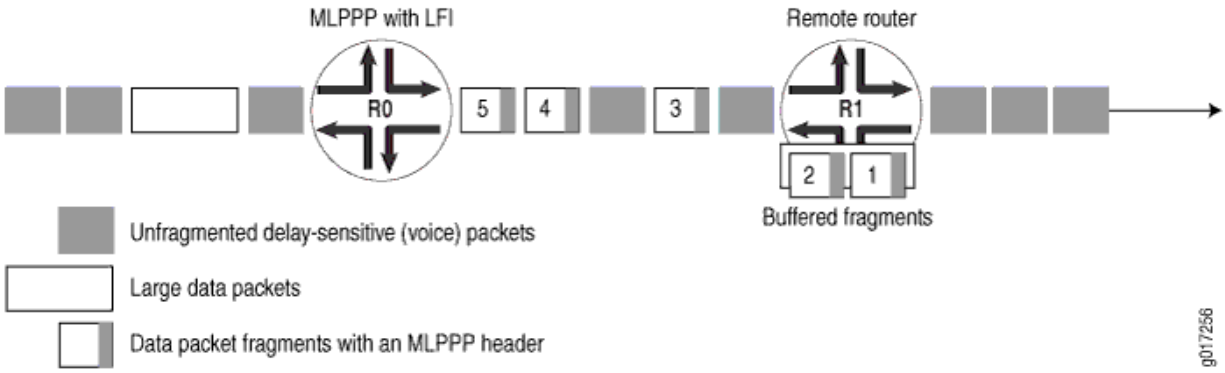
Link fragmentation and interleaving (LFI) solves this problem. It reduces delay and *jitter* on links by fragmenting large packets and interleaving delay-sensitive packets with the resulting smaller packets for simultaneous transmission across multiple links of a multilink bundle.

Figure 19 on page 370 illustrates how LFI works. In this figure, device R0 and device R1 have LFI enabled. When device R0 receives large and small packets, such as data and voice packets, it divides them into two categories. All voice packets and any other packets configured to be treated as voice packets are categorized as LFI packets and transmitted without fragmentation or an MLPPP header. If CRTP is configured on the bundle, LFI packets are transmitted through CRTP processing. The remaining non-LFI (data) packets can be fragmented or unfragmented based on the configured fragmentation threshold. The packets larger than the fragmentation threshold are fragmented. An MLPPP header (containing a multilink sequence number) is added to all non-LFI packets, fragmented and unfragmented.

The fragmentation is performed according to the fragmentation threshold that you configure. For example, if you configure a fragmentation threshold of 128 bytes, all packets larger than 128 bytes are fragmented. When device R1 receives the packets, it sends the unfragmented voice packets immediately but buffers the packet fragments until it receives the last fragment for a packet. In this example, when device R1 receives fragment 5, it reassembles the fragments and transmits the whole packet.

The unfragmented data packets are treated as a single fragment. Thus device R1 does not buffer the unfragmented data packets and transmits them as it receives them.

Figure 19: LFI on a Services Router



To configure LFI, you define the MLPPP encapsulation type and enable fragmentation and interleaving of packets by specifying the fragmentation threshold and fragmentation maps, with a no-fragmentation knob mapped to the forwarding class of choice.

Example: Configuring Link Fragmentation and Interleaving

IN THIS SECTION

This example shows how to configure LFI.

Requirements

Before you begin, you should have two Juniper Networks devices configured with at least two serial interfaces that communicate over serial links. This example shows two devices.

Overview

In this example, you create an interface called lsq-0/0/0. You specify the encapsulation type as multilink-ppp and set the fragmentation threshold value to 128. Set a fragmentation threshold of 128 bytes on the MLPPP bundle so that it applies to all traffic on both constituent links, enabling that any

packet larger than 128 bytes transmitted on these links is fragmented. Any nonzero value must be a multiple of 64 bytes. The value can be between 128 and 16320. The default value is 0 bytes.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

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

To configure LFI:

1. Create an interface.

```
[edit]
user@host# edit interfaces lsq-0/0/0
```

2. Specify the encapsulation type and fragmentation threshold value.

```
[edit interfaces lsq-0/0/0]
user@host# set unit 0 encapsulation multilink-ppp fragment-threshold 128
```

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

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

Verification

IN THIS SECTION

Verifying Link Fragmentation and Interleaving Configuration

Purpose

Verify the LFI configuration.

Action

From operational mode, enter the `show interfaces lsq-0/0/0` command.

Configuring Class-of-Service on Link Services Interfaces

IN THIS SECTION

On a Juniper Networks device, when LFI is enabled, all forwarding traffic assigned to queue 2 or member link is treated as LFI (voice) traffic. The topics below discuss the overview of classifiers and

forwarding class, definition and application of schedule maps, and overview and configuration details of interface shaping rates.

Understanding How to Define Classifiers and Forwarding Classes

By defining classifiers you associate incoming packets with a forwarding class and loss priority. Based on the associated forwarding class, you assign packets to output queues. To configure classifiers, you specify the bit pattern for the different types of traffic. The classifier takes this bit pattern and attempts to match it to the type of packet arriving on the interface. If the information in the packet's header matches the specified pattern, the packet is sent to the appropriate queue, defined by the forwarding class associated with the classifier.

On a Juniper Networks device, when LFI is enabled, all forwarding traffic assigned to queue 2 or member link is treated as LFI (voice) traffic. You do not need to assign network control traffic to a queue explicitly, because it is assigned to queue 3 by default.

NOTE: On member links:

- `DATA` is assigned to queue 0.
- `VOICE` is assigned to queue 2.
- `NC` (network control) is assigned to queue 3. By default NC is assigned to queue 3.

Example: Defining Classifiers and Forwarding Classes

IN THIS SECTION

This example shows how to define classifiers for different types of traffic, such as voice, data, and network control packets, and to direct the traffic to different output queues to manage your throughput.

Requirements

Before you begin:

- Configure two Juniper Networks devices with at least two serial interfaces that communicate over serial links.
- Configure CoS components. See *Junos OS Class of Service Configuration Guide for Security Devices*.

Overview

In this example, you configure class of service and set the default IP precedence classifier to classify_input, which is assigned to all incoming traffic. You then set the precedence bit value in the type of service field to 000 for all incoming data traffic and 010 for all incoming voice traffic. You set all outgoing data traffic to queue 0 and all voice traffic to queue 2, and fragmentation-map maps queue 2 to no fragmentation.

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

```
set class-of-service classifiers inet-precedence classify_input forwarding-class DATA loss-
priority low code-points 000
set class-of-service classifiers inet-precedence classify_input forwarding-class VOICE loss-
priority low code-points 010
set class-of-service forwarding-classes queue 0 DATA
set class-of-service forwarding-classes queue 2 VOICE
set class-of-service forwarding-classes queue 3 NC
set class-of-service interfaces ge-0/0/1 unit 0 classifiers inet-precedence classify_input
```

```
set class-of-service fragmentation-maps FM forwarding-class VOICE no-fragmentation
set class-of-service interfaces lsq-0/0/0 unit 0 fragmentation-map FM
```

Step-by-Step Procedure

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

To define classifiers and forwarding classes:

1. Configure class of service.

```
[edit]
user@host# edit class-of-service
```

2. Configure the behavior aggregate classifier for classifying packets.

```
[edit class-of-service]
user@host# edit classifiers inet-precedence classify_input
```

3. Assign packets with IP precedence to the data forwarding class and specify a loss priority.

```
[edit class-of-service classifiers inet-precedence classify_input]
user@host# set forwarding-class DATA loss-priority low code-points 000
```

4. Assign packets with IP precedence to the voice forwarding class and specify a loss priority.

```
[edit class-of-service classifiers inet-precedence classify_input]
user@host# set forwarding-class VOICE loss-priority low code-points 010
```

5. Specify the forwarding class one-to-one with the output queues.

```
[edit class-of-service]
user@host# edit forwarding-classes
user@host# set queue 0 DATA
```

```
user@host# set queue 2 VOICE
user@host# set queue 3 NC
```

6. Create an interface and apply the behavior aggregate classifier.

```
[edit class-of-service]
user@host# edit interfaces ge-0/0/1
user@host# set unit 0 classifiers inet-precedence classify_input
```

7. Configure fragmentation map.

```
[edit]
user@host# edit class-of-service
user@host# set fragmentation-maps FM forwarding-class VOICE no-fragmentation
```

8. Attach fragmentation map to the interface.

```
[edit class-of-service]
user@host# set interfaces lsq-0/0/0 unit 0 fragmentation-map FM
```

Results

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

```
[edit]
    user@host# show class-of-service
classifiers {
        inet-precedence classify_input {
            forwarding-class DATA {
        loss-priority low code-points 000;
        }
            forwarding-class VOICE {
        loss-priority low code-points 010;
        }
            }
        }
```

```
        forwarding-classes {
            queue 0 DATA;
            queue 2 VOICE;
            queue 3 NC;
            }
            interfaces {
                lsq-0/0/0 {
                unit 0 {
                fragmentation-map FM;
                    }
                    }
                    ge-0/0/1 {
                        unit 0 {
                        classifiers {
                    inet-precedence classify_input;
                    }
                }
            }
            }
            fragmentation-maps {
                FM {
                forwarding-class {
                VOICE {
                    no-fragmentation;
                }
            }
        }
    }
```

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

Verification

IN THIS SECTION

- Verifying Classifiers and Forwarding Classes | **378**

To confirm that the configuration is working properly, perform this task:

Verifying Classifiers and Forwarding Classes

Purpose

Verify the classifiers and the forwarding classes.

Action

From operational mode, enter the `show class-of-service` command.

Understanding How to Define and Apply Scheduler Maps

Juniper Networks devices support per-unit scheduling `set class-of-service schedulers S0 priority low`, which allows you to configure scheduler maps on each MLPPP or MLFR multilink bundle. You can also configure scheduler maps on constituent links, but you must maintain the same relative priority on the constituent links and on the multilink bundle.

If you configure CoS components with LFI on a Juniper Networks device, we recommend that you follow certain recommendations for shaping rate, scheduling priority, and buffer size.

When you configure LFI, we recommend that you configure the shaping rate on each constituent link of the multilink bundle. Shaping rate configuration on the constituent links is required to limit the *jitter* on the LFI queue. If you anticipate no delay-sensitive or jitter-sensitive traffic on the LFI queue, or if there is no LFI traffic at all, shaping rate configuration is optional.

Table 51 on page 378 shows an example of correct and incorrect relative priorities on a multilink bundle and its constituent link. In this example, you have assigned a high priority to LFI packets and a low priority to data packets on the multilink bundle. To maintain the relative priority on the constituent links, you can assign a high priority to the LFI packets and a medium-high priority to the data packets, but you cannot assign a medium-high priority to LFI packets and a high priority to data packets.

Table 51: Relative Priorities on Multilink Bundles and Constituent Links

Multilink Bundle	Correct Constituent Link Priorities	Incorrect Constituent Link Priorities
LFI packets—High priority	LFI packets—High priority	LFI packet—Medium-high priority
Data packets—Low priority	Data packets—Medium-high priority	Data packets—High priority

By defining schedulers you configure the properties of output queues that determine the transmission service level for each queue. These properties include the amount of interface bandwidth assigned to the queue, the size of the memory buffer allocated for storing packets, and the priority of the queue. After defining schedulers you associate them with forwarding classes by means of scheduler maps. You then associate each scheduler map with an interface, thereby configuring the hardware queues and packet schedulers that operate according to this mapping.

NOTE: When data and LFI streams are present, the following scheduler map configuration is recommended for constituent links. This gives less latency for LFI traffic and avoids out-of-order transmission of data traffic.

Configure the following schedulers:

- `set class-of-service schedulers S0 buffer-size temporal 20k`
- `set class-of-service schedulers S0 priority low`
- `set class-of-service schedulers S2 priority high`
- `set class-of-service schedulers S3 priority high`

Configure the following scheduler map:

- `set class-of-service scheduler-maps lsqlink_map forwarding-class best-effort scheduler S0`
- `set class-of-service scheduler-maps lsqlink_map forwarding-class assured-forwarding scheduler S2`
- `set class-of-service scheduler-maps lsqlink_map forwarding-class network-control scheduler S3`

Attach scheduler map to all member links:

- `set class-of-service interfaces t1-2/0/0 unit 0 scheduler-map lsqlink_map`

NOTE: Even after this configuration, if out-of-range sequence number drops are observed on the reassembly side, increase the drop-timeout of the bundle to 200 ms.

Example: Configuring Scheduler Maps

IN THIS SECTION

This example shows how to configure scheduler maps to determine the transmission service level for each output queue.

Requirements

Before you begin, you should have two Juniper Networks devices configured with at least two serial interfaces that communicate over serial links.

Overview

IN THIS SECTION

In this example, you create interfaces called lsq-0/0/0, se-1/0/0, and se-1/0/1. You enable per-unit scheduling to allow the configuration of scheduler maps on the bundle. You configure a scheduler map as s_map on lsq-0/0/0. You then apply the scheduler map to the constituent links, se-1/0/0 and se-1/0/1, of the multilink bundle. You associate the scheduler with each of the forwarding classes, DATA, VOICE and NC. You define the properties of output queues for the DATA scheduler by setting the transmit rate and the buffer size to 49 percent. You specify the properties of output queues for the VOICE scheduler by setting the transmit rate to 50 percent, the buffer size to 5 percent, and the priority to high. Finally, you define the properties of output queues for the NC scheduler by setting the transmit rate and the buffer size to 1 percent and the priority to high.

Topology

Configuration

IN THIS SECTION

- Procedure | 381

Procedure

CLI Quick Configuration

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

```
set interfaces lsq-0/0/0 per-unit-scheduler
set interfaces se-1/0/0 per-unit-scheduler
set interfaces se-1/0/1 per-unit-scheduler
set class-of-service interfaces lsq-0/0/0 unit 0 scheduler-map s_map
set class-of-service interfaces se-1/0/0 unit 0 scheduler-map s_map
set class-of-service interfaces se-1/0/1 unit 0 scheduler-map s_map
set class-of-service scheduler-maps s_map forwarding-class DATA scheduler DATA
set class-of-service scheduler-maps s_map forwarding-class VOICE scheduler VOICE
set class-of-service scheduler-maps s_map forwarding-class NC scheduler NC
set class-of-service schedulers DATA transmit-rate percent 49
set class-of-service schedulers DATA buffer-size percent 49
set class-of-service schedulers VOICE transmit-rate percent 50
set class-of-service schedulers VOICE buffer-size percent 5
set class-of-service schedulers VOICE priority high
set class-of-service schedulers NC transmit-rate percent 1
set class-of-service schedulers NC buffer-size percent 1
set class-of-service schedulers NC priority high
```

Step-by-Step Procedure

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

To configure scheduler maps:

1. Create interfaces and enable per-unit scheduling.

```
[edit interfaces]
user@host# set lsq-0/0/0 per-unit-scheduler
user@host# set se-1/0/0 per-unit-scheduler
user@host# set se-1/0/1 per-unit-scheduler
```

2. Define a scheduler map and apply it to the constituent links in the multilink bundle.

```
[edit class-of-service interfaces]
user@host# set lsq-0/0/0 unit 0 scheduler-map s_map
user@host# set se-1/0/0 unit 0 scheduler-map s_map
user@host# set se-1/0/1 unit 0 scheduler-map s_map
```

3. Associate a scheduler with each forwarding class.

```
[edit class-of-service scheduler-maps]
user@host# set s_map forwarding-class DATA scheduler DATA
user@host# set s_map forwarding-class VOICE scheduler VOICE
user@host# set s_map forwarding-class NC scheduler NC
```

4. Define the properties of output queues for the DATA scheduler.

```
[edit class-of-service schedulers]
user@host# set DATA transmit-rate percent 49
user@host# set DATA buffer-size percent 49
```

5. Define the properties of output queues for the VOICE scheduler.

```
[edit class-of-service schedulers]
user@host# set VOICE transmit-rate percent 50
user@host# set VOICE buffer-size percent 5
user@host# set VOICE priority high
```

6. Define the properties of output queues for the NC scheduler.

```
[edit class-of-service schedulers]
user@host# set NC transmit-rate percent 1
user@host# set NC buffer-size percent 1
user@host# set NC priority high
```

Results

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

```
[edit]
    user@host# show class-of-service
    interfaces {
    lsq-0/0/0 {
     unit 0 {
        scheduler-map s_map;
    }
    }
    se-1/0/0 {
        unit 0 {
        scheduler-map s_map;
    }
    }
    se-1/0/1 {
        unit 0 {
        scheduler-map s_map;
        }
    }
    }
        scheduler-maps {
    s_map {
    forwarding-class DATA scheduler DATA;
        forwarding-class VOICE scheduler VOICE;
        forwarding-class NC scheduler NC;
    }
}
    schedulers {
        DATA {
```

```
        transmit-rate percent 49;
        buffer-size percent 49;
        }
        VOICE {
            transmit-rate percent 50;
            buffer-size percent 5;
            priority high;
            }
            NC {
                transmit-rate percent 1;
                buffer-size percent 1;
            priority high;
        }
    }
```

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

Verification

IN THIS SECTION

- Verifying the Configuration of scheduler maps. | 384

To confirm that the configuration is working properly, perform this task:

Verifying the Configuration of scheduler maps.

Purpose

Verify the configuration of scheduler maps.

Action

From operational mode, enter the `show class-of-services lsq-0/0/0 scheduler-map s_map`, `show class-of-services se-1/0/0 scheduler-map s_map`, and `show class-of-services se-1/0/1 scheduler-map s_map` commands.

Understanding Interface Shaping Rates

When you configure LFI, we recommend that you configure the shaping rate on each constituent link of the multilink bundle. Shaping rate configuration on the constituent links is required to limit the *jitter* on the LFI queue. If you anticipate no delay-sensitive or jitter-sensitive traffic on the LFI queue, or if there is no LFI traffic at all, shaping rate configuration is optional.

The shaping rate specifies the amount of bandwidth to be allocated for the multilink bundle. You must configure the shaping rate to be equal to the combined physical interface bandwidth for the constituent links. The combined bandwidth capacity of the two constituent links is 2 Mbps. Hence, configure a shaping rate of 2 Mbps on each constituent link.

Example: Configuring Interface Shaping Rates

IN THIS SECTION

This example shows how to configure interface shaping rates to control the maximum rate of traffic transmitted on an interface.

Requirements

Before you begin:

- Configure two Juniper Networks devices configured with at least two serial interfaces that communicate over serial links. For more information about serial interfaces. See Serial Interfaces Overview.
- To apply shaping rates to interfaces, you have to first enable per-unit scheduling. For more information on per-unit scheduling. See "Example: Configuring Scheduler Maps" on page 380.

Overview

IN THIS SECTION

In this example, you set the shaping rate to 2000000 for the constituent links of the multilink bundle, se-1/0/0 and se-1/0/1.

Topology

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To configure the interface shaping rates:

1. Configure class of service.

```
[edit]
user@host# edit class-of-service
```

2. Apply the shaping rates to the constituent links of the multilink bundle.

```
[edit class-of-service]
user@host# set interfaces se-1/0/0 unit 0 shaping-rate 2000000
user@host# set interfaces se-1/0/1 unit 0 shaping-rate 2000000
```

Verification

To verify the configuration is working properly, enter the `show class-of-service` command.

Achieving Greater Bandwidth, Load Balancing, and Redundancy with Multilink Bundles

IN THIS SECTION

The topics below discuss the overview of MLPPP Bundles and Link Fragmentation and Interleaving (LFI) on Serial Links, and configuring an MLPP bundle on security devices.

Understanding MLPPP Bundles and Link Fragmentation and Interleaving (LFI) on Serial Links

Juniper Networks devices support MLPPP and MLFR multilink encapsulations. MLPPP multilink encapsulation enables you to bundle multiple PPP links into a single multilink bundle and MLFR multilink encapsulation enables you to bundle multiple Frame Relay data-link connection identifiers (DLCIs) into a single multilink bundle. Multilink bundles provide additional bandwidth, load balancing, and redundancy by aggregating low-speed links, such as T1, E1, and serial links.

NOTE: Currently, Junos OS supports bundling of only one xDSL link under bundle interface.

You configure multilink bundles as logical units or channels on the link services interface `lsq-0/0/0`:

- With MLPPP and MLFR FRF.15, multilink bundles are configured as logical units on `lsq-0/0/0`—for example, `lsq-0/0/0.0` and `lsq-0/0/0.1`.

- With MLFR FRF.16, multilink bundles are configured as channels on `lsq-0/0/0`—for example, `lsq-0/0/0:0` and `lsq-0/0/0:1`.

After creating multilink bundles, you add constituent links to the bundle. The constituent links are the low-speed physical links that are to be aggregated. You can create 64 multilink bundles, and on each multilink bundle you can add up to 8 constituent links. The following rules apply when you add constituent links to a multilink bundle:

- On each multilink bundle, add only interfaces of the same type. For example, you can add either T1 or E1, but not both.
- Only interfaces with a PPP encapsulation can be added to an MLPPP bundle, and only interfaces with a Frame Relay encapsulation can be added to an MLFR bundle.
- If an interface is a member of an existing bundle and you add it to a new bundle, the interface is automatically deleted from the existing bundle and added to the new bundle.

Configuring a multilink bundle on the two serial links increases the bandwidth by 70 percent from approximately 1 Mbps to 1.7 Mbps and prepends each packet with a multilink header as specified in the FRF.12 standard. To increase the bandwidth further, you can add up to eight serial links to the bundle. In addition to a higher bandwidth, configuring the multilink bundle provides load balancing and redundancy. If one of the serial links fails, traffic continues to be transmitted on the other links without any interruption. In contrast, independent links require routing policies for load balancing and redundancy. Independent links also require IP addresses for each link as opposed to one IP address for the bundle. In the routing table, the multilink bundle is represented as a single interface.

Example: Configuring an MLPPP Bundle

IN THIS SECTION

This example shows how to configure an MLPPP bundle to increase traffic bandwidth.

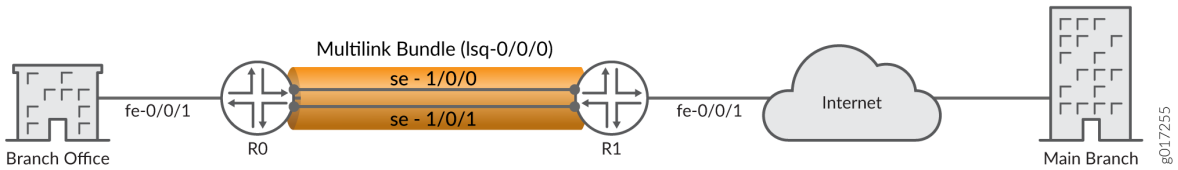
Requirements

Before you begin, you should have two Juniper Networks devices configured with at least two serial interfaces that communicate over serial links.

Overview

In this example, you create the MLPPP bundle lsq-0/0/0.0 at the logical unit level of the link services interface lsq-0/0/0 on Juniper Networks devices R0 and R1. You then add the two serial interfaces se-1/0/0 and se-1/0/1 as constituent links to the multilink bundle. In Figure 20 on page 389, your company's branch office is connected to its main branch using devices R0 and R1. You transmit data and voice traffic on two low-speed 1-Mbps serial links. To increase bandwidth, you configure MLPPP and join the two serial links se-1/0/0 and se-1/0/1 into the multilink bundle lsq-0/0/0.0. Then you configure LFI and CoS on R0 and R1 to enable them to transmit voice packets ahead of data packets.

Figure 20: Configuring MLPPP and LFI on Serial Links



Configuration

IN THIS SECTION

- Procedure | **390**

Procedure

CLI Quick Configuration

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

```
For device R0
set interfaces lsq-0/0/0 unit 0 family inet address 10.0.0.10/24
set interfaces se-1/0/0 unit 0 family mlppp bundle lsq-0/0/0.0
set interfaces se-1/0/1 unit 0 family mlppp bundle lsq-0/0/0.0
set interfaces se-1/0/0 serial-options clocking-mode dce clock-rate 2.0mhz
set interfaces se-1/0/1 serial-options clocking-mode dce clock-rate 2.0mhz
```

```
For device R1
set interfaces lsq-0/0/0 unit 0 family inet address 10.0.0.9/24
set interfaces se-1/0/0 unit 0 family mlppp bundle lsq-0/0/0.0
set interfaces se-1/0/1 unit 0 family mlppp bundle lsq-0/0/0.0
```

Step-by-Step Procedure

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

To configure MLPPP bundle:

1. Create an interface on both devices.

```
[edit]
user@host# edit interfaces lsq-0/0/0 unit 0
```

2. Configure a family inet and define the IP address on device R0.

```
[edit interfaces lsq-0/0/0 unit 0]
user@host# set family inet address 10.0.0.10/24
```

3. Configure a family inet and define the IP address on device R1.

```
[edit interfaces lsq-0/0/0 unit 0]
user@host# set family inet address 10.0.0.9/24
```

4. Specify the names of the constituent links to be added to the multilink bundle on both devices.

```
[edit interfaces]
user@host# edit se-1/0/0 unit 0
user@host# set family mlppp bundle lsq-0/0/0.0
[edit interfaces]
user@host# edit se-1/0/1 unit 0
user@host# set family mlppp bundle lsq-0/0/0.0
```

5. Set the serial options to the same values for both interfaces on R0.

NOTE: R0 is set as a DCE device. The serial options are not set for interfaces on R1. You can set the serial options according to your network setup.

```
[edit interfaces]
user@host# set se-1/0/0 serial-options clocking-mode dce clock-rate 2.0mhz
user@host# set se-1/0/1 serial-options clocking-mode dce clock-rate 2.0mhz
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces lsq-0/0/0`, `show interfaces se-1/0/0`, and `show interfaces se-1/0/1` commands for R0 and R1. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
For device R0
[edit]
    user@host# show interfaces lsq-0/0/0
family inet {
    address 10.0.0.10/24;
        }
    }
[edit]
```

```
user@host# show interfaces se-1/0/0
    clocking-mode dce;
        clock-rate 2.0mhz;
        }
        unit 0 {
            family mlppp {
            bundle lsq-0/0/0.0;
        }
    }
[edit]
    user@host# show interfaces se-1/0/1
serial-options {
    clocking-mode dce;
        clock-rate 2.0mhz;
        }
        unit 0 {
            family mlppp {
            bundle lsq-0/0/0.0;
        }
    }
```

```
For device R1
[edit]
user@host# show interfaces lsq-0/0/0
    family inet {
        address 10.0.0.9/24;
        }
    }
[edit]
    user@host# show interfaces se-1/0/0
    unit 0 {
    family mlppp {
        bundle lsq-0/0/0.0;
    }
}
[edit]
    user@host# show interfaces se-1/0/1
    unit 0 {
    family mlppp {
        bundle lsq-0/0/0.0;
```

```
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the MLPPP Bundle

Purpose

Verify that the constituent links are added to the bundle correctly.

Action

From operational mode, enter the `show interfaces lsq-0/0/0 statistics` command.

Configuring Compressed Real-Time Transport Protocol

IN THIS SECTION

Compressed Real-Time Transport Protocol (CRTP) is typically used for compressing voice and video packets. The topics below discuss the overview of CRTP and its configuration details.

Understanding Compressed Real-Time Transport Protocol

Compressed Real-Time Transport Protocol (CRTP) is typically used for compressing voice and video packets. You can configure CRTP with LFI on a link services interface.

CRTP can be configured as a compression device on a T1 or E1 interface with PPP encapsulation, using the link services interface.

NOTE:

- F-max period—Maximum number of compressed packets allowed between transmission of full headers. It has a range from 1 to 65,535.
- Maximum and Minimum—UDP port values from 1 to 65,536 reserve these ports for RTP compression. CRTP is applied to network traffic on ports within this range. This feature is applicable only to voice services interfaces.

Example: Configuring the Compressed Real-Time Transport Protocol

IN THIS SECTION

This example shows how to configure CRTP to improve packet transmission, especially for time-sensitive voice packets.

Requirements

Before you begin, you should have two Juniper Networks devices configured with at least two serial interfaces that communicate over serial links.

Overview

In this example, you create a T1 interface called t1-1/0/0 and set the type of encapsulation to PPP. You set the link services intelligent queuing interface to lsq-0/0/0.0. You then create an interface called lsq-0/0/0 and set the logical unit 0. Finally, you set the F-max period to 2500, the minimum UDP port value to 2000, and the maximum UDP port value to 64009.

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

```
set interfaces t1-1/0/0 encapsulation ppp
set interfaces t1-1/0/0 unit 0 compression-device lsq-0/0/0.0
set interfaces lsq-0/0/0 unit 0 compression rtp f-max-period 2500 port minimum 2000 maximum
64009
```

Step-by-Step Procedure

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

To configure CRTP on a device:

1. Create the T1 interface.

```
[edit]
user@host# edit interfaces t1-1/0/0
```

2. Set the type of encapsulation.

```
[edit interfaces t1-1/0/0]
user@host# set encapsulation ppp
```

3. Add the link services intelligent queuing interface to the physical interface.

```
[edit interfaces t1-1/0/0]
user@host# edit unit 0
user@host# set compression-device lsq-0/0/0.0
```

4. Create an interface and set the logical unit.

```
[edit interfaces]
user@host# edit lsq-0/0/0 unit 0
```

5. Configure the link services intelligent queuing interface.

```
[edit interfaces lsq-0/0/0 unit 0]
user@host# set compression rtp f-max-period 2500 port minimum 2000 maximum 64009
```

Results

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

```
[edit]
    user@host# show interfaces
    lsq-0/0/0 {
    unit 0 {
        compression {
```

```
        rtp {
        f-max-period 2500;
            port minimum 2000 maximum 64009;
            }
        }
    }
}
    t1-1/0/0 {
        encapsulation ppp;
    unit 0 {
        compression-device lsq-0/0/0.0;
            }
        }
```

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

Verification

IN THIS SECTION

- Verifying the CRTP Configuration | 397

Confirm that the configuration is working properly.

Verifying the CRTP Configuration

Purpose

Verify the CRTP configuration.

Action

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

RELATED DOCUMENTATION

Link Services Interfaces Overview | 336

CHAPTER 6

Configuring Management, Discard, and Loopback Interfaces

IN THIS CHAPTER

Configuring Management and Discard Interfaces

IN THIS SECTION

The topics below discuss the over and configuration details of management and discard interfaces on the security devices.

Configuring Management Interfaces

Management interfaces are the primary interfaces for accessing the device remotely. Typically, a management interface is not connected to the in-band network, but is connected instead to the device's internal network. Through a management interface you can access the device over the network using utilities such as `ssh` and `telnet` and configure it from anywhere, regardless of its physical location. SNMP can use the management interface to gather statistics from the device.

Management interfaces vary based on device type:

- The SRX5600 and SRX5800 devices include a 10/100-Mbps Ethernet port on the Routing Engine (RE). This port, which is labeled ETHERNET, is a dedicated out-of-band management interface for the device. Junos OS automatically creates the device's management interface `fxp0`. To use `fxp0` as a management port, you must configure its logical port `fxp0.0` with a valid IP address. While you can use `fxp0` to connect to a management network, you cannot place it into the management zone.

NOTE: On the SRX5600 and SRX5800 devices, you must first connect to the device through the serial console port before assigning a unique IP address to the management interface.

As a security feature, users cannot log in as `root` through a management interface. To access the device as `root`, you must use the console port.

In an SRX Series Firewall, the `fxp0` management interface is a dedicated port located on the Routing Engine. In an SRX Series chassis cluster configuration, the control link interface must be port `0` on an

SPC. For each node in the chassis cluster, you must configure the SPC that is used for the control link interface.

Configuring Discard Interface

The discard (dsc) interface is not a physical interface, but a virtual interface that discards packets. You can configure one discard interface. This interface allows you to identify the ingress (inbound) point of a denial-of-service (DoS) attack. When your network is under attack, the target host IP address is identified, and the local policy forwards attacking packets to the discard interface. Traffic routed out the discard interface is silently discarded.

RELATED DOCUMENTATION

Understanding Interfaces | 2

Configuring Loopback Interfaces

IN THIS SECTION

The topics below discuss the overview and configuration details of loopback interfaces on security devices.

Loopback Interface Overview

The Internet Protocol (IP) specifies a loopback network with the (IPv4) address 127.0.0.0/8. Most IP implementations support a loopback interface (lo0) to represent the loopback facility. Any traffic that a computer program sends on the loopback network is addressed to the same computer. The most

commonly used IP address on the loopback network is `127.0.0.1` for IPv4 and `::1` for IPv6. The standard domain name for the address is `localhost`.

A network device also includes an internal loopback interface (`lo0.16384`). The internal loopback interface is a particular instance of the loopback interface with the logical unit number 16384.

You use the loopback interface to identify the device. While you can use any interface address to determine if the device is online, the loopback address is the preferred method. Whereas interfaces might be removed or addresses changed based on network topology changes, the loopback address never changes.

When you ping an individual interface address, the results do not always indicate the health of the device. For example, a subnet mismatch in the configuration of two endpoints on a point-to-point link makes the link appear to be inoperable. Pinging the interface to determine whether the device is online provides a misleading result. An interface might be unavailable because of a problem unrelated to the device configuration or operation. You can use the loopback interface to address these issues.

Junos OS Evolved supports two different filters to control the flow of local packets: one for network control traffic (loopback traffic) and one for management traffic. For additional information, see Top Differences Between Junos OS Evolved and Junos OS.

Benefits

- As the loopback address never changes, it is the best way to identify a device in the network.
- The loopback interface is always up and reachable as long as the route to that IP address is available in the IP routing table. Hence, you can use the loopback interface for diagnostics and troubleshooting purposes.
- Protocols such as OSPF use the loopback address to determine protocol-specific properties for the device or network. Further, some commands such as `ping mpls` require a loopback address to function correctly.
- Junos OS creates a separate loopback interface for the internal routing instance, which prevents any filter on `lo0.0` from disrupting internal traffic.

Configuring a Loopback Interface

The loopback interface supports many different network and operational functions and is an *always-up* interface. This means that the loopback interface ensures that the device is reachable, even if some of the physical interfaces are down or removed, or an IP address has changed. In most cases, you always define a loopback interface.

Junos OS follows the IP convention of identifying the loopback interface as lo0.

Junos OS requires that the loopback interface always be configured with a /32 network mask because the Routing Engine is essentially a host.

If you are using routing instances, you can configure the loopback interface for the default routing instance or for a specific routing instance. The following procedure adds the loopback interface to the default routing instance.

Optionally, instead of configuring the loopback interface at the `[edit interfaces]` hierarchy level, you can use a configuration group, as shown in this procedure. This is a recommended best practice for configuring the loopback interface. This procedure uses a group called `global` as an example.

To configure a loopback interface:

1. Using the host IP address, assign it to the loopback interface.

   Each host in your network deployment should have a unique loopback interface address. The address used here is only an example.

   ```
   [edit groups global interfaces lo0 unit 0 family inet]
   user@host# set address 192.0.2.27/32
   ```

2. (Optional) Set the preferred IP address.

   You can configure as many addresses as you need on the lo0 interface, so it is good practice to designate one preferred IP address.

   ```
   [edit groups global interfaces lo0 unit 0 family inet]
   user@host# set address 192.0.2.48/32 preferred
   ```

3. (Optional) Configure additional addresses.

   Only unit 0 is permitted as the primary loopback interface. If you want to add more IP addresses to unit 0, you configure them in the normal way under unit 0, without the `preferred` option.

   ```
   [edit groups global interfaces lo0 unit 0 family inet]
   user@host# set address 198.51.100.48/32
   user@host# set address 192.168.11.27
   ```

   NOTE: You do not have to include the `/32` as long as the IPv4 address is a valid host address. (This usually means that the last octet cannot be zero.)

4. Configure the localhost address.

On the lo0.0 interface, it is useful to have the IP address 127.0.0.1 configured, as certain processes such as NTP and MPLS ping use this default host address. The 127.0.0.1/32 address is a Martian IP address (an address invalid for routing), so it is never advertised by the Juniper Networks device.

```
[edit groups global interfaces lo0 unit 0 family inet]
user@host# set address 127.0.0.1/32
```

5. (Optional) Configure an ISO address.

   Depending on your network configuration, you might also need an ISO address for the IS-IS routing protocol.

```
[edit groups global interfaces lo0 unit 0 family iso]
user@host# address 49.0026.0000.0000.0110.00
```

6. If you used a configuration group, apply the configuration group, substituting `global` with the appropriate group name.

```
[edit]
user@host# set apply-groups global
```

7. Commit the configuration.

```
user@host# commit
```

CHAPTER 7

LTE Mini-PIM

IN THIS CHAPTER

LTE Mini Physical Interface Modules (LTE Mini-PIM)

SUMMARY

Learn about the LTE MIni-PIM, the features supported on it and how to configure it on security devices.

IN THIS SECTION

LTE Mini-PIM Overview

IN THIS SECTION

The LTE Mini-Physical Interface Module (Mini-PIM) provides wireless WAN support on security devices. Table 52 on page 405 specifies the key details of the LTE Mini-PIM interface.

Table 52: LTE Mini-PIM Device Details

Interface Details	Descriptions
Interface name	LTE Mini-PIM
Supported on	For information about platforms support, see hardware compatibility tool (HCT).
Models	• SRX-MP-LTE-AE • SRX-MP-LTE-AA See LTE Mini-PIM Models.

Table 52: LTE Mini-PIM Device Details *(Continued)*

Interface Details	Descriptions

Physical interface for the 4G LTE Mini-PIM	• The interface name is `cl-`*`slot number`*`/0/0` where *slot number* identifies the slot on the device in which you insert the LTE Mini-PIM. For example, `cl-1/0/0`.
	• Configurable properties on the physical interface are:
	  • A dialer pool to which the physical interface belongs and the priority of the interface in the pool.
	  • Profiles for the SIM cards.
	  • Radio access technology (automatic, 3G, LTE).

Key deployment	• Provides wireless WAN support. • Operates on 3G and 4G networks.

For hardware specifications for the LTE Mini-PIM, see LTE Mini-Physical Interface Module.

Features Supported on the LTE Mini-PIM

Table 53 on page 406 describes the key features supported on LTE Mini-PIM.

Table 53: Key Features Supported on LTE Mini-PIM

Feature	Description
Automatic switchover between service providers through dual SIMs	Supports dual Subscriber Identity Module (SIM) cards that allow connectivity to two different ISP networks. Automatic switchover provides a failover mechanism when the current active network fails.
Multiple service provider and Access Point Name (APN) profiles	Supports up to 16 profiles configuration for each SIM. The LTE Mini-PIM supports two SIM cards, you can configure a total of 32 profiles and at a time, only single profile is active.
SIM security functions	Supports security functions such as SIM lock and unlock, and PIN change.

Table 53: Key Features Supported on LTE Mini-PIM *(Continued)*

Feature	Description
Primary, logical and backup interface with always-on, dial-on-demand, and backup modes	On receiving traffic, the logical dl0 interface enables and places calls through the physical interface in the dialer pool. The dialer interface performs backup and dialer filter functions. You can configure the dialer interface to operate as: • Primary Interface: The dialer interface connects to the network and is always on. For more information, see Configuring the LTE Mini-PIM as the Primary Interface. • Backup interface for the primary WAN connection: The dialer interface activates only when the primary connection fails. For more information, see Configuring the LTE Mini-PIM as a Backup Interface. • Dial-on-demand: For more information, see Configuring the LTE Interface as a Dial-on-Demand Interface. Configuration modes: always-on, dial-on-demand or backup modes. You can configure the Mini-PIM in any one of the modes. • Always-on: The Mini-PIM connects to the 3G/4G network after booting. The connection is always maintained. • When you configure as primary interface, the LTE Mini-PIM supports both the always-on and dial-on-demand modes.
Over-the-air upgrade for modem firmware	Supports Over-the-air (OTA) firmware upgrade that enables automatic and timely upgrade of modem firmware when new firmware versions are available. You can enable or disable the OTA upgrade on the LTE Mini-PIM. OTA upgrade is disabled by default.

Configure LTE Mini-PIM

IN THIS SECTION

You can configure the LTE Mini-PIM as a primary interface, as a backup interface or as a dial-on-demand interface.

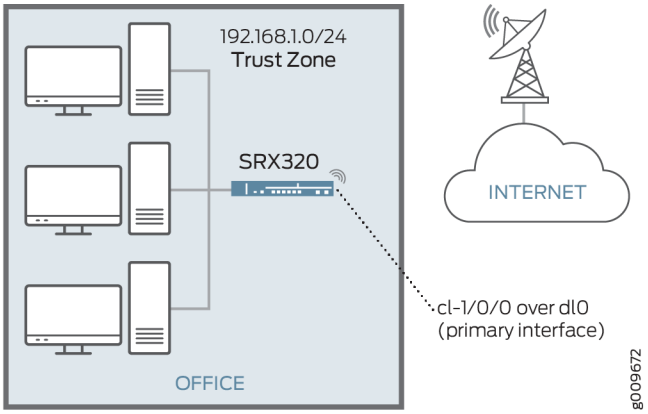
Configure LTE Mini-PIM as a Primary Interface

Before you begin, ensure that dl0.0 is not configured as a backup interface. If dl0.0 is configured as a backup for any interface on the SRX Series Firewall, then this configuration overrides the configuration outlined in this procedure, and the LTE Mini-PIM will function as a backup interface.

Use the `show interfaces | display set | match backup-option | match dl0.0` command to check whether any interface uses dl0.0 as a backup interface. If dl0.0 is configured as a backup interface, then delete the configuration by issuing the following command:`delete interfaces` *`interface-name`* `unit 0 backup-options interface dl0.0`

The LTE Mini-PIM is installed on a SRX320 line of devices and functions as the primary interface as seen in Figure 1 and assumed that the LTE Mini-PIM is installed in slot 1 on the SRX320 line of devices.

Figure 21: LTE Mini-PIM as a Primary Interface



To configure the LTE Mini-PIM as a primary interface:

1. Configure the dialer interface.

```
user@host# set interfaces dl0 unit 0 family inet negotiate-address
user@host# set interfaces dl0 unit 0 family inet6 negotiate-address
user@host# set interfaces dl0 unit 0 dialer-options pool dialer-pool-number
user@host# set interfaces dl0 unit 0 dialer-options dial-string dial-number
user@host# set interfaces dl0 unit 0 dialer-options always-on
```

2. Configure the dialer pool for the LTE Mini-PIM physical interface.

```
user@host# set interfaces cl-1/0/0 dialer-options pool number
```

3. Configure the profile for the Subscriber Identity Module (SIM) cards.

```
user@host# run request modem wireless create-profile profile-id profile-id cl-1/0/0 slot sim-slot-number
access-point-name apn-name authentication-method none
```

`sim-slot-number` is the slot on the Mini-PIM in which the SIM card is inserted.

4. Verify that the profile is configured successfully.

```
user@host# run show modem wireless profiles cl-1/0/0 slot 1
```

5. Activate the SIM card.

```
user@host# set interfaces cl-1/0/0 act-sim sim-slot-number
```

6. Select the profile and configure the radio access type for the SIM card.

```
user@host# set interfaces cl-1/0/0 cellular-options sim sim-slot-number select-profile profile-id profile-id
user@host# set interfaces cl-1/0/0 cellular-options sim sim-slot-number radio-access automatic
```

   If a SIM card is installed in the second slot, then select the profile and configure the radio access type for the secondary SIM card.

7. Verify the status of the wireless network and dialer interface.

```
user@host#  run show modem wireless network
user@host#  run show interfaces dl0.0
```

   If the LTE Mini-PIM gets an IP address with a mask of /32 from the service provider, you can configure the default gateway information using the **set interfaces *cl-interface* cellular-options sim *sim-slot* gateway *ip-address/mask*** command to make the Mini-PIM accept the assigned IP address.

Configure LTE Mini-PIM in a High Availability Cluster Mode

An SRX chassis cluster supports two `cl` interfaces, `cl-1/1/0` (primary node) and `cl-8/1/0` (secondary node).

To configure the LTE Mini-PIM in a HA cluster mode:

1. Configure the dialer interface (`dl0`).

```
{primary:node0}[edit]
user@host# set interfaces dl0 unit 0 family inet negotiate-address
user@host# set interfaces dl0 unit 0 family inet6 negotiate-address
user@host# set interfaces dl0 unit 0 dialer-options pool dialer-pool-number
user@host# set interfaces dl0 unit 0 dialer-options dial-string dial-number
user@host# set interfaces dl0 unit 0 dialer-options always-on
```

2. Configure the LTE interface (`cl-1/1/0`) on the primary node.

a. Configure the dialer pool for the LTE physical interface.

```
{primary:node0}[edit]
user@host# set interfaces cl-1/1/0 dialer-options pool dialer-pool-number
```

b. Specify the priority for the interface. The interface with the higher priority becomes the active interface.

```
{primary:node0}[edit]
user@host# set interfaces cl-1/1/0 dialer-options pool dialer-pool-number priority priority
```

c. Configure the profile for the SIM cards.

```
{primary:node0}[edit]
user@host# run request modem wireless create-profile profile-id profile-id cl-1/1/0 slot
sim-slot-number access-point-name apn-name
```

d. Verify that the profile is configured successfully.

```
user@host# run show modem wireless profiles cl-1/1/0 slot 1
```

e. Activate the SIM card.

```
{primary:node0}[edit]
user@host# set interfaces cl-1/1/0 act-sim sim-slot-number
```

f. Select the profile and configure the radio access type for the SIM card.

```
{primary:node0}[edit]
user@host# set interfaces cl-1/1/0 cellular-options sim sim-slot-number select-profile
profile-id profile-id
user@host# set interfaces cl-1/1/0 cellular-options sim sim-slot-number radio-access
automatic
```

3. Repeat Step 2 to configure the LTE interface (`cl-8/1/0`) for the secondary node.

   If you assign the same priority to both interfaces, then the interface that is listed first in the configuration becomes the active interface.

Verify the active interface:

```
root@host> show dialer pools
Pool: 1
  Dialer interfaces:      Name            State
                          dl0.0           Active
  Subordinate interfaces: Name            Flags         Priority
                          cl-1/1/0        Active        100
                          cl-8/1/0        Inactive      1
```

4. Verify the status of the wireless network and dialer interface.

```
user@host# run show modem wireless network
user@host#  run show interfaces dl0.0
```

By default, the time interval taken to switch to the secondary cl interface when the active cl interface times out is 120 seconds. You can change the time interval by configuring the `redial-delay` option.

```
{primary:node0}[edit]
user@host# user@host# set interfaces dl0 unit 0 dialer-options redial-delay time-in-seconds
```

5. Verify that the profile is configured successfully.

```
user@host# run show modem wireless profiles cl-1/0/0 slot 1
```

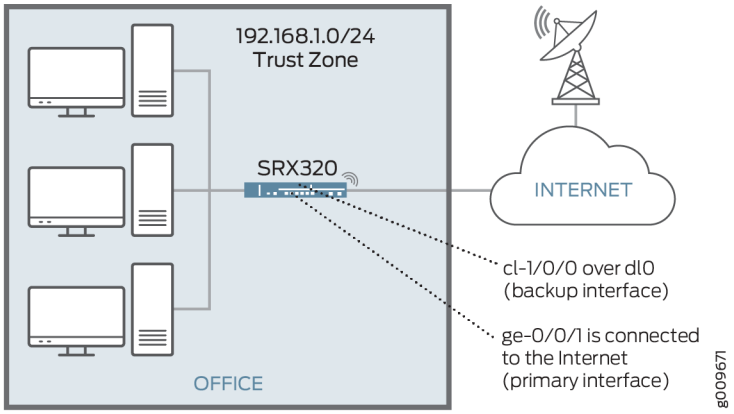
6. Verify the status of the wireless network and dialer interface.

```
user@host#  run show modem wireless network
user@host#  run show interfaces dl0.0
```

Configure LTE Mini-PIM as a Backup Interface

You can configure the LTE Mini-PIM as a backup interface. If the primary interface fails, the LTE Mini-PIM connects to the network and remains online only until the primary interface becomes functional. The dialer interface is enabled only when the primary interface fails. LTE Mini-PIM installed on SRX320 and functions as a backup interface as shown in Figure 22 on page 413. The `ge-0/0/1` port is connected to the internet and functions as the primary interface. In this scenario, the Mini-PIM is installed on slot 1.

Figure 22: LTE Mini-PIM as a Backup Interface



To configure the LTE Mini-PIM as a backup interface:

1. Configure the dialer interface.

```
user@host# set interfaces dl0 unit 0 family inet negotiate-address
user@host# set interfaces dl0 unit 0 family inet6 negotiate-address
user@host# set interfaces dl0 unit 0 dialer-options pool dialer-pool-number
user@host# set interfaces dl0 unit 0 dialer-options dial-string dial-number
```

2. Configure the dialer pool for the LTE Mini-PIM physical interface.

```
user@host# set interfaces cl-1/0/0 dialer-options pool dialer-pool-number
```

3. Configure the profile for the SIM cards.

   `sim-slot-number` is the slot on the Mini-PIM in which the SIM card is inserted.

```
user@host# run request modem wireless create-profile profile-id profile-id cl-1/0/0 slot sim-slot-number
access-point-name l3vpn.corp authentication-method none
```

4. Verify that the profile is configured successfully.

```
user@host# run show modem wireless profiles cl-1/0/0 slot 1
```

5. Activate the SIM card.

```
user@host# set interfaces cl-1/0/0 act-sim sim-slot-number
```

6. Select the profile and configure the radio access type for the SIM card.

```
user@host# set interfaces cl-1/0/0 cellular-options sim sim-slot-number select-profile profile-id profile-id
user@host# set interfaces cl-1/0/0 cellular-options sim sim-slot-number radio-access automatic
```

7. Configure the Ethernet interface as the primary interface, which connects to the wireless network. Configure the dl0 interface as the backup interface.

```
user@host# set interfaces ge-0/0/1 unit 0 family inet address 192.168.2.1/24
user@host# set interfaces ge-0/0/1 unit 0 backup-options interface dl0.0
```

8. Verify the status of the wireless network and dialer interface.

```
user@host#  run show modem wireless network
user@host#  run show interfaces dl0.0
```

You can use the `activation-delay` and `deactivation-delay` command-line options to avoid interface flaps. Avoid the Interface flaps by forcing a delay between the time the primary interface changes states, and the time the dialer interface is enabled or disabled. The activation delay controls the time between the primary interface going down and activation of the dialer interface. Similarly, the deactivation delay controls the time between the recovery of the primary interface and deactivation of the backup interface.

You can insert the SIM of another LTE provider in the second SIM slot if there is an issue with the active SIM (for example, weak signal). The second SIM now becomes the active SIM.

The switchover between the two SIMs is automatic and no manual control is involved. This automatic switchover only happens when there is an issue with the active SIM (the active SIM is removed or has a weak signal). The active SIM tries to re-connect 3 times and in case of failure, the other SIM becomes active and starts connecting.

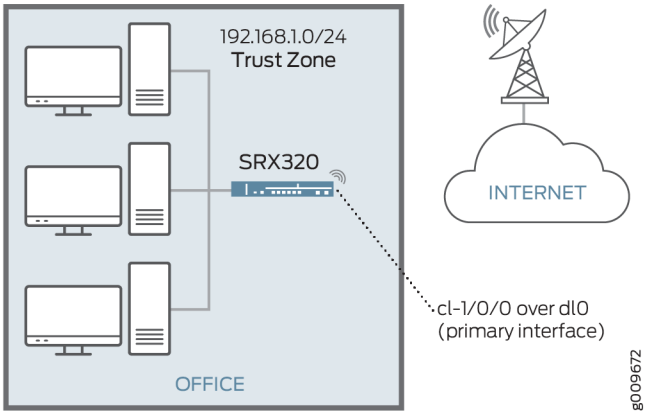
Configure LTE Mini-PIM as a Dial-on-demand Interface

When you configure the LTE interface as a primary interface, it functions either in always-on or in dial-on-demand mode. In always-on mode, the interface remains connected to the network whereas in dial-on-demand mode, the connection is established only when needed.

In dial-on-demand mode, you can enable the dialer interface only when network traffic configured as an “interesting traffic” arrives on the network. Interesting traffic triggers or activates the wireless WAN connection. You define an interesting packet by using the dialer filter. To configure dial-on-demand by using a dialer filter, you first configure the dialer filter and then apply the filter to the dialer interface. Once the traffic is sent over the network, an inactivity timer is triggered and the connection is closed after the timer expires. The dial-on-demand mode is supported only if the LTE Mini-PIM is configured as a primary interface.

The LTE Mini-PIM installed on an SRX320 functions as the primary interface as show in Figure 23 on page 415 and assumed that the LTE Mini-PIM is installed in slot 1 on the device.

Figure 23: LTE Mini-PIM as a Dial-on-Demand Interface



To configure the LTE Mini-PIM as a dial-on-demand interface:

1. Configure the dialer interface.

```
user@host# set interfaces dl0 unit 0 family inet negotiate-address
user@host# set interfaces dl0 unit 0 family inet6 negotiate-address
user@host# set interfaces dl0 unit 0 family inet filter dialer dialer-filter-name
user@host# set interfaces dl0 unit 0 dialer-options pool dialer-pool-number
user@host# set interfaces dl0 unit 0 dialer-options dial-string dial-number
```

   Optionally, you can configure the `idle-timeout` value, to determine the duration of the enabled connection in the absence of interesting traffic.

```
user@host# set interfaces dl0 unit 0 dialer-options idle-timeout idle-timeout-value
```

2. Configure the dialer pool for the LTE Mini-PIM physical interface.

```
user@host# set interfaces cl-1/0/0 dialer-options pool number
```

3. Create the dialer filter rule.

```
user@host# set firewall family inet dialer-filter dialer-filter-name term term1 from destination-
address ip-address then note
```

4. Set the default route.

```
set routing-options static route ip-address next-hop dl0.0
```

5. Configure the profile for the SIM cards.

```
user@host# run request modem wireless create-profile profile-id profile-id cl-1/0/0 slot sim-slot-
number access-point-name apn-name authentication-method none
```

6. Verify that the profile is configured successfully.

```
user@host# run show modem wireless profiles cl-1/0/0 slot 1
```

7. Activate the SIM card.

```
user@host# set interfaces cl-1/0/0 act-sim sim-slot-number
```

8. Select the profile and configure the radio access type for the SIM card.

```
user@host# set interfaces cl-1/0/0 cellular-options sim sim-slot-number select-profile profile-id
profile-id
user@host# set interfaces cl-1/0/0 cellular-options sim sim-slot-number radio-access automatic
```

9. Verify the configuration by sending traffic to the destination address. The traffic is routed to the dl0 interface and if it matches the dialer filter rule, then the dl0 is triggered to dial.

10. Verify the status of the wireless network and dialer interface.

```
user@host#  run show modem wireless network
user@host#  run show interfaces dl0.0
```

Example: Configure LTE Mini-PIM as a Backup Interface

IN THIS SECTION

This example shows how to configure the LTE Mini-PIM as a backup interface. If the primary interface fails, the Mini-PIM connects to the network and remains online only until the primary interface becomes functional. The dialer interface is enabled only when the primary interface fails. In this scenario, the Mini-PIM is installed on slot 1.

Requirements

Overview

Configuration

IN THIS SECTION

CLI Quick Configuration

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

```
set interfaces dl0 unit 0 family inet negotiate-address
set interfaces dl0 unit 0 family inet6 negotiate-address
set interfaces dl0 unit 0 dialer-options pool dialer-pool-number
set interfaces dl0 unit 0 dialer-options dial-string dial-number
set interfaces cl-1/0/0 dialer-options pool dialer-pool-number
run request modem wireless create-profile profile-id profile-id cl-1/0/0 slot sim-slot-number
access-point-name l3vpn.corp authentication-method none
run show modem wireless profiles cl-1/0/0 slot 1
set interfaces cl-1/0/0 act-sim sim-slot-number
set interfaces cl-1/0/0 cellular-options sim sim-slot-number select-profile profile-id profile-id
set interfaces cl-1/0/0 cellular-options sim sim-slot-number radio-access automatic
set interfaces ge-0/0/1 unit 0 family inet address 192.168.2.1/24
set interfaces ge-0/0/1 unit 0 backup-options interface dl0.0
```

Configure the LTE Mini-PIM as a Backup Interface

Step-by-Step Procedure

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

To configure LTE Mini-PIM as a backup interface:

1. Create the dialer interface.

```
[edit interfaces]
user@host# set interfaces dl0 unit 0 family inet negotiate-address
user@host# set interfaces dl0 unit 0 family inet6 negotiate-address
user@host# set interfaces dl0 unit 0 dialer-options pool dialer-pool-number
user@host# set interfaces dl0 unit 0 dialer-options dial-string dial-number
```

2. Define the dialer pool for the LTE Mini-PIM physical interface.

```
user@host# set interfaces cl-1/0/0 dialer-options pool dialer-pool-number
```

3. Create and configure the profile on the SIM cards.

   `sim-slot-number` is the slot on the Mini-PIM in which the SIM card is inserted.

```
user@host# run request modem wireless create-profile profile-id profile-id cl-1/0/0 slot sim-slot-number access-point-name l3vpn.corp authentication-method none
```

4. Activate the SIM card.

```
user@host# set interfaces cl-1/0/0 act-sim sim-slot-number
```

5. Select the profile and configure the radio access type for the SIM card.

```
user@host# set interfaces cl-1/0/0 cellular-options sim sim-slot-number select-profile profile-id profile-id
user@host# set interfaces cl-1/0/0 cellular-options sim sim-slot-number radio-access automatic
```

6. Specify Ethernet interface as the primary interface, which connects to the wireless network. Specify the dl0 interface as the backup interface.

```
user@host# set interfaces ge-0/0/1 unit 0 family inet address 192.168.2.1/24
user@host# set interfaces ge-0/0/1 unit 0 backup-options interface dl0.0
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces dl0.0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host> show interfaces dl0.0
Logical interface dl0.0 (Index 353) (SNMP ifIndex 559)
    Flags: Up Point-To-Point SNMP-Traps 0x4004000 Encapsulation: ENET2
    Dialer:
```

```
    State: Active, Dial pool: pool1
    Primary interface: ge-1/0/1.0 (Index 350)
    Dial strings: 1234
    Subordinate interfaces: cl-1/1/0 (Index 161)
    Activation delay: 0, Deactivation delay: 0
    Initial route check delay: 120
    Redial delay: 120
    Callback wait period: 5
    Load threshold: 0, Load interval: 60
    Input packets : 7
    Output packets: 10
    Protocol inet, MTU: 1490
    Max nh cache: 0, New hold nh limit: 0, Curr nh cnt: 0, Curr new hold cnt: 0, NH drop cnt: 0
    Flags: Sendbcast-pkt-to-re, Negotiate-Address
    Addresses, Flags: Is-Preferred Is-Primary
    Destination: 100.100.60.208/29, Local: 100.100.60.212, Broadcast: 100.100.60.215
    Protocol inet6, MTU: 1490
    Max nh cache: 75000, New hold nh limit: 75000, Curr nh cnt: 0, Curr new hold cnt: 0, NH drop
cnt: 0
    Flags: Is-Primary, Negotiate-Address
    Addresses, Flags: Is-Preferred
    Destination: fe80::/64, Local: fe80::5a00:bb0f:fcaa:7d00
```

Verification

IN THIS SECTION

Verification of the configured profile

Purpose

Verify that the profile is configured successfully.

Action

From operational mode, run the `show modem wireless profiles cl-1/0/0 slot 1` command.

```
user@host> show modem wireless profiles cl-1/0/0 slot 1

Profile details
  Max profiles: 16
  Default profile Id: 1

Profile 1: ACTIVE
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4V6
Profile 2: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 3: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 4: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 5: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 6: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 7: Inactive
  Valid: TRUE
```

```
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 8: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 9: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 10: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 11: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 12: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 13: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 14: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
Profile 15: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
```

```
Profile 16: Inactive
  Valid: TRUE
  Access point name (APN): airtelgprs.com
  Authentication: None
  IP Version: IPV4
```

Meaning

The output confirms the profile is active.

Verification of status of the dialer interface

Purpose

Verify that the dialer interface is configured successfully.

Action

From operational mode, run the `show interfaces dl0.0` command.

```
user@host> show interfaces dl0.0

    Logical interface dl0.0 (Index 353) (SNMP ifIndex 559)
    Flags: Up Point-To-Point SNMP-Traps 0x4004000 Encapsulation: ENET2
    Dialer:
    State: Active, Dial pool: pool1
    Primary interface: ge-1/0/1.0 (Index 350)
    Dial strings: 1234
    Subordinate interfaces: cl-1/1/0 (Index 161)
    Activation delay: 0, Deactivation delay: 0
    Initial route check delay: 120
    Redial delay: 120
    Callback wait period: 5
    Load threshold: 0, Load interval: 60
    Input packets : 7
    Output packets: 10
    Protocol inet, MTU: 1490
    Max nh cache: 0, New hold nh limit: 0, Curr nh cnt: 0, Curr new hold cnt: 0, NH drop cnt: 0
    Flags: Sendbcast-pkt-to-re, Negotiate-Address
    Addresses, Flags: Is-Preferred Is-Primary
    Destination: 100.100.60.208/29, Local: 100.100.60.212, Broadcast: 100.100.60.215
```

```
    Protocol inet6, MTU: 1490
    Max nh cache: 75000, New hold nh limit: 75000, Curr nh cnt: 0, Curr new hold cnt: 0, NH drop
cnt: 0
    Flags: Is-Primary, Negotiate-Address
    Addresses, Flags: Is-Preferred
    Destination: fe80::/64, Local: fe80::5a00:bb0f:fcaa:7d00
```

Meaning

The output confirms the interface `dlo` is configured and active.

Verification of status of the modem network and modem firmware

Purpose

Verify that the wireless network is configured, check the firmware, and check if the sim is active.

Action

From operational mode, enter the `show modem wireless network cl-1/0/0` command to verify the network status and `show modem wireless firmware cl-1/0/0` command to verify the firmware and sim status. Alternatively you can use the `show configuration` command to verify the complete status.

```
user@host> show modem wireless network cl-1/0/0

LTE Connection details
Connected time: 147
IP: 172.16.52.4
Gateway: 172.16.52.5
DNS: 123.123.123.123
Input bps: 0
Output bps: 0
Bytes Received: 1308
Bytes Transferred: 1164
Packets Received: 10
Packets Transferred: 10
Wireless Modem Network Info
Current Modem Status: Connected
Current Service Status: Normal
Current Service Type: PS
Current Service Mode: LTE
```

```
Current Band: B3
Network: UNICOM
Mobile Country Code (MCC): 460
Mobile Network Code (MNC): 1
Location Area Code (LAC): 65534
Routing Area Code (RAC): 0
Cell Identification: 4865903
Access Point Name (APN): abcde
Public Land Mobile Network (PLMN): CHN-UNICOM
Physical Cell ID (PCI): 333
International Mobile Subscriber Identification (IMSI): ***************
International Mobile Equipment Identification (IMEI/MEID): ***************
Integrate Circuit Card Identity (ICCID): 89860114721100697502
Reference Signal Receiving Power (RSRP): -97
Reference Signal Receiving Quality (RSRQ): -16
Signal to Interference-plus-Noise Ratio (SiNR): 0
Signal Noise Ratio (SNR): 0
Energy per Chip to Interference (ECIO): 0
```

Meaning

The output here shows the wireless modem network is connected and IP address of the fimware connected.

RELATED DOCUMENTATION

dialer-options

8
CHAPTER

Wi-Fi MPIM

IN THIS CHAPTER

Wi-Fi Mini Physical Interface Module (MPIM)

IN THIS SECTION

The Wi-Fi Mini-Physical Interface Module (Mini-PIM) provides an integrated wireless access point (or wireless LAN) solution along with routing, switching, and security in a single device. The topics below describes the overview and configuration of Wi-Fi Mini-PIM.

Wi-Fi Mini-Physical Interface Module Overview

IN THIS SECTION

Mini-PIM supports the 802.11ac Wave 2 wireless standards and is backward compatible with 802.11a/b/g/n. You can use the three new models of the Wi-Fi Mini-PIM based on the regional wireless standard requirements;

- SRX-MP-WLAN-US — The model based on USA's wireless standard.
- SRX-MP-WLAN-IL — The model based on Israel's wireless standard.
- SRX-MP-WLAN-WW — The model for other countries.

You cannot change the country code for the SRX-MP-WLAN-US and SRX-MP-WLAN-IL models as they are fixed. The Wi-Fi Mini-PIM can coexist with other Mini-PIMs. Table 54 on page 429 provides a summary of the features supported on Mini-PIM.

Typical deployments for Wi-Fi Mini-PIM solution include:

- Secure wireless LAN connectivity to endpoint devices of corporate users at remote branch offices. 802.11ac, WPA2, 802.1X, and SSID-to-VLAN mapping features provide secure Wireless LAN connectivity.
- Direct network connectivity to the enterprise Internet of Things (IoT) devices. The security features secures the IoT devices.

See How to Install the Wi-Fi Mini-PIM.

Wireless LAN Interface in Chassis Cluster Mode

The Mini-PIM is also supported in chassis cluster mode to provide redundancy. Wireless users are connected to the active interface in redundancy group. To support chassis cluster mode for wireless LAN interface Mini-PIM, you need to configure chassis cluster setup with two wireless LAN interfaces `wl-`*x*`/0/0` and `wl-`*y*`/0/0`, where *x* indicates the slot number which wireless LAN interface Mini-PIM plug in on the node 0 and *Y* indicates the slot number which wireless LAN interface Mini-PIM plug in on the node 1.

In chassis cluster mode, there is one wireless LAN interface active, the other wireless LAN interface is inactive. Wi-Fi client is associated to active wireless LAN interface.

Below are the list of events which trigger wireless LAN interface failover when:

- wireless LAN interface is abnormal.
- primary wireless LAN interface is down.
- Redundant group which wireless LAN interface belongs to failover manually.
- primary WLAN interface node is failed.

After wireless LAN interface failover, the original inactive wireless LAN interface is changed to active and the Wi-Fi client sessions are reconnected to the new primary wireless LAN interface.

With chassis cluster mode, WLAND process runs on both nodes. The WLAND on primary node pushes the WLAN configuration to PFE on two nodes, and then PFE forwards the configuration to local wireless LAN interface card so that two wireless LAN interface cards have the same configuration.

To monitor wireless LAN interface status, WLAND finds the wireless LAN interface to be abnormal, it can trigger redundant group failover. In Layer 3 mode, by default, wireless LAN interface activity monitor is configured for WLAN high availability using the commands `set chassis cluster redundancy-group`

`1 interface-monitor wl-2/0/0 weight 255` and `set chassis cluster redundancy-group 1 interface-monitor wl-7/0/0 weight 255`.

The new primary wireless LAN interface is active and the abnormal wireless LAN interface card is restarted and goes to inactive state. The Wi-Fi client is reconnected to the active wireless LAN interface automatically since the configuration (radio, channel, bandwidth, ssid, and so on) on active WAP is same as the original wireless LAN interface.

Wireless LAN Interface in Layer 3 (L3) Mode

The interfaces are configured as subordinate interface of RETH using the command `set interfaces wl-x/0/0 gigether-options redundant-parent reth-interface`. You can add the RETH interface to one redundant group and set the priority for each node in the redundant group. Only one wireless LAN interface is active in the redundant group and the other one is inactive.

Wireless LAN Interface in Layer 2 (L2) Mode

You can build a chassis cluster with wireless LAN interface Mini-PIM. The peer wireless LAN interfaces are configured in the same VLAN and the wireless LAN interface on the primary node of redundant group zero is chosen as active interface by default. L2 mode (`family ethernet-switching`) of wireless LAN interface behave like any other L2 switching port (trunk port).

Features Supported on the Wi-Fi Mini-PIM

Table 54 on page 429 lists the key features supported on the Wi-Fi Mini-PIM.

Table 54: Wi-Fi Mini-PIM Features

Feature	Description
2x2 MU-MIMO	Enables transmission of data to multiple clients simultaneously.
Dual radios	Both radios of 2.4 GHz and 5 GHz bands are simultaneously supported. The maximum supported speed is up to 1.2 Gbps.

Table 54: Wi-Fi Mini-PIM Features *(Continued)*

Feature	Description
Virtual access points (VAPs) and VLAN features	• Allows you to segment the WLAN into multiple broadcast domains that are the wireless equivalents of Ethernet VLANs. A single access point is segregated into multiple individual VAPs, simulating multiple access points in a single system. • An access point supports multiple VLANs, which can be distributed across VAPs and radios. • You can configure up to eight VAPs per radio. You can map up to 16 extended service set identifiers (ESSIDs) to individual VLANs. • The VLANs from the Mini-PIM software map to VLANs on Junos OS.
Co-existence of interfaces	The Wi-Fi Mini-PIM coexists with 4G LTE, VDSL, T1, and serial interfaces.
Client authentication methods	Client authentication methods supported are Wi-Fi Protected Access (WPA) Enterprise (WPA2 standards) and Wi-Fi Protected Access (WPA) Personal (AES-CCMP cipher suits and WPA2 standards).

Configure Wi-Fi Mini-PIM

IN THIS SECTION

You can configure the radios and virtual access points on the Wi-Fi Mini-PIM. This topic contains sections that describe the basic Wi-Fi Mini-PIM configuration at the wireless interface level. For more information about how to install a Wi-Fi Mini-PIM, see How to Install the Wi-Fi Mini-PIM.

The following sections describe how to configure the Wi-Fi Mini-PIM.

Configure Network Setting for the Wi-Fi Mini-PIM

Configure wl- interface

The interface name for the Mini-PIM is denoted as `wl-x/0/0`, where *x* is the slot on the device in which the Mini-PIM is installed. The wl- interface is created automatically when you insert the Mini-PIM into the slot on the device.

To configure the wireless LAN interface:

1. Configure an IP address for the Wi-Fi interface:

```
[edit interfaces]
user@host# set interfaces wl-x/0/0 unit unit-number family inet address address
```

2. Configure the address pool.

```
[edit]
user@host# set access address-assignment pool pool-name family inet network ip-address
user@host# set access address-assignment pool pool-name family inet range range-name low ip-address
user@host# set access address-assignment pool pool-name family inet range range-name high ip-address
user@host# set access address-assignment pool pool-name family inet dhcp-attributes router router ip-address
```

   The DHCP address pool and the Wi-FI interface must be in the same network.

3. Enable the DHCP server on the interface.

```
[edit interfaces]
user@host# set system services dhcp-local-server group group interface wl-x/0/0
```

   The eth0 interface on the Mini-PIM enables the DHCP client. If the DHCP server is enabled on the wl interface, the server assigns an IP address to the eth0 interface. You can view the binding information by issuing the `show dhcp server binding` command.

4. Assign the interface to a security zone.

```
[edit interfaces]
user@host# set security zones security-zone zone interface wl-x/0/0
```

Configure Access Point

To configure the access point associated with the wireless LAN interface wl-*x*/0/0:

1. Configure the interface.

```
[edit]
user@host# set wlan access-point name interface wl-x/0/0
```

2. Set the country code (applicable only for SRX-MP-WLAN-WW models of the Mini-PIM).

> **NOTE**: If you do not set the country code for the SRX-MP-WLAN-WW models, the Mini-PIM considers the country code as US. You cannot set the country code for the SRX-MP-WLAN-US and SRX-MP-WLAN-IL models.

```
[edit]
user@host# set wlan access-point name access-point-options country country-code
```

3. Set the physical location (location of your hardware device, example: 1st-floor).

```
[edit]
user@host# set wlan access-point name location location
```

4. Commit the configuration.

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

Configure Radios

Every access point has two radios—radio 1 operates at 5-GHz bandwidth and radio 2 operates at 2.4-GHz bandwidth. A VAP is configured based on the radio. You can configure up to eight VAPs per radio and map up to 16 ESSIDs to individual VLANs. Wi-Fi Mini-PIM supports both the radios (2.4 and 5 GHz) to work simultaneously. You can also disable a radio. Table 2 lists the modes supported on each radio.

Changing the radio settings can cause the access point to stop and restart system processes. If this occurs, wireless clients that are connected to the access point temporarily lose connectivity. We recommend that you change radio settings when WLAN traffic is low.

Table 55: Supported Modes on Wi-Fi Mini-PIM Radios

Radio	Supported Modes
Radio 1 (5.0 GHz)	• an—802.11a and 802.11n clients operating on 5 GHz frequency can connect to the access point • acn—802.11a, 802.11n and 802.11ac clients operating on 5 GHz frequency can connect to the access point
Radio 2 (2.4 GHz)	• gn—802.11g, 802.11b and 802.11n clients operating in 2.4 GHz frequency can connect to the access point. This is the default mode for this radio. • g—802.11g clients operating in 2.4 GHz frequency can connect to the access point supported from Junos OS Release 20.4R1.

To configure the radio:

1. Configure the radio mode. Radio 1 supports acn and an modes. Radio 2 supports only gn mode.

```
For radio 1:
[edit]
user@host# set wlan access-point name radio 1 radio-options mode [an|acn]
```

```
For radio 2:
[edit]
user@host# set wlan access-point name radio 2 radio-options mode gn
```

2. Configure the channel number. If you select auto, then the Mini-PIM chooses the channel automatically. By default, channel number is set to `auto`.

```
[edit]
user@host# set wlan access-point name radio [1|2] radio-options channel number [auto /
channel-number]
```

3. Configure the channel bandwidth. The default channel bandwidth is 20 MHz for the 2.4 GHz radio and 40 MHz for the 5 GHz radio. You can only set 80 MHz as the channel bandwidth for 5 GHz radio and not for 2.4GHz.

```
[edit]
user@host# set wlan access-point name radio [1|2] radio-options channel bandwidth [20|40|80]
```

4. Configure the transmit power. You can configure the transmit power on a per-radio basis.

> **NOTE**: When you configure the transmit power, the Mini-PIM card will fix transmit power to the specified value set, in this case, the power by rate functionality does not work. So it is recommended not to set transmit power to a specified value. When you do not configure the transmit power (do not fix the transmit power to a specified value), the power by rate functionality works. If you configure the transmit power percentage to 100, then it chooses the option "auto", the behavior is similar to no transmit power configured and power by rate functionality will work.

```
[edit]
user@host# set wlan access-point name radio [1|2] radio-options transmit-power percent
```

5. Commit the configuration.

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

In countries where Dynamic Frequency Selection (DFS) is required, the Wi-Fi card performs appropriate checks for radar. DFS is enabled by default. If you set the `channel number` to `auto`, the access point selects the channel from the list of DFS and non-DFS channels. You can disable DFS by using the `dfs-off` option **set wlan access-point *name* radio 1 radio-options dfs-off**.

Only the 5 GHz radio (radio 1) supports DFS.

For more information on DFS, see Channels and Frequencies Supported on the Wi-Fi Mini-PIM.

Configure Virtual Access Points (VAP)

VAPs allow segmentation of the wireless LAN into multiple broadcast domains that are the wireless equivalents of Ethernet VLANs. To configure the VAP:

1. Enter an ID and description for the VAP.

```
[edit]
user@host# set wlan access-point name radio [1|2] virtual-access-point id description
description
```

2. Enter the SSID value.

```
[edit]
user@host# set wlan access-point name radio [1|2] virtual-access-point id ssid ssid
```

3. Configure one of the following security authentication methods for the VAP.

   - none—The data transferred between clients and the access point is not encrypted. Clients can associate with the access point without any authentication.

     ```
     [edit]
     user@host# set wlan access-point name radio [1|2] virtual-access-point id security none
     ```

   - wpa-enterprise—The device authenticates through an 802.1X-compliant RADIUS server.

     ```
     [edit]
     user@host# set wlan access-point name radio [1|2] virtual-access-point id security wpa-
     enterprise cipher-suites ccmp
     user@host# set wlan access-point name radio [1|2] virtual-access-point id security wpa-
     enterprise radius-server ip-address
     user@host# set wlan access-point name radio [1|2] virtual-access-point id security wpa-
     enterprise radius-port port
     user@host# set wlan access-point name radio [1|2] virtual-access-point id security wpa-
     enterprise radius-key secret-key
     user@host# set wlan access-point name radio [1|2] virtual-access-point id security wpa-
     enterprise wpa-version v2
     ```

- wpa-personal—The device uses preshared keys (PSKs) or a passphrase for authentication and encryption. Keys are stored on the device and on all wireless clients. You do not need to configure a separate authentication server.

```
[edit]
user@host# set wlan access-point name radio [1|2] virtual-access-point id security wpa-personal cipher-suites ccmp
user@host# set wlan access-point name radio [1|2] virtual-access-point id security wpa-personal key-type [ascii|hex]
user@host# set wlan access-point name radio [1|2] virtual-access-point id security wpa-personal key password
user@host# set wlan access-point name radio [1|2] virtual-access-point id security wpa-personal wpa-version v2
```

4. Configure and specify the upload and download rate limits on the Wi-Fi Mini-PIM. The range for `upload-limit` and `download-limit` is from 256 Kbps to 1,048,576 Kbps.

```
[edit]
user@host# set wlan access-point name radio [1|2] virtual-access-point id upload-limit upload-limit-rate
user@host# set wlan access-point name radio [1|2] virtual-access-point id download-limit download-limit-rate
```

5. Specify the maximum number of clients that can be connected to the VAP.

```
[edit]
user@host# set wlan access-point name radio [1|2] virtual-access-point id maximum-stations number
```

6. Commit the configuration.

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

After completing the configuration successfully, you can view the parameters by using the `show wlan access-points name detail` command.

Configure VLANS

Configure VLANs based on VAP

(Optional) A single access point is segregated into multiple individual virtual access points (VAPs) simulating multiple access points in a single system. The access point supports multiple VLANs. To configure the VLAN ID based on the VAP:

1. Configure the VLAN for the wireless LAN interface (wl- interface). Follow the below steps to configure VLAN ID based on the VAP :

```
[edit]
user@host# set vlans vlan-name vlan-id vlan-id
user@host# set vlans vlan-name vlan-id-list vid-list
user@host# set interfaces wl-x/0/0 unit unit-number family ethernet-switching vlan members
all
```

2. Set trunk mode on the wl- interface.

```
[edit]
user@host# set interfaces wl-x/0/0 unit unit-number family ethernet-switching interface-mode
trunk
```

3. Set the native VLAN of the wl- interface.

```
[edit]
user@host# set interfaces wl-x/0/0 native-vlan-id vlan-id
```

When you configure native vlan, the wl- interface will add a tag when it receives an untagged packet and takes no action when it receives a tagged native-vlan-id packet.

4. Configure the access point for the wl- interface.

```
[edit]
user@host# set wlan access-point name interface wl-x/0/0
```

5. Configure all VAP parameters including the radio mode, channel number, and VAP SSID, VAP VLAN ID on the Wi-Fi Mini-PIM.

```
[edit]
user@host# set wlan access-point name radio (1| 2) radio-options mode (an| gn | acn)
user@host# set wlan access-point name radio (1| 2) radio-options channel number  (auto |
```

```
channel-number)
user@host# set wlan access-point name radio (1| 2) virtual-access-point id ssid ssid
user@host# set wlan access-point name radio (1| 2) virtual-access-point id vlan vlan-id
```

6. Commit the configuration.

Configure WPA enterprise authentication

(Optional) Wi-Fi protected access (WPA) enterprise is Wi-Fi alliance standard that uses RADIUS server authentication with AES-CCMP cipher suite. With this mode you can use high security encryption along with a centrally managed user authentication. Only the WPA2 standard is supported. To configure the WPA enterprise authentication:

1. Configure the address book and assign a security zone.

```
[edit]
user@host# set security address-book book-name address address-name ip-prefix
user@host# set security address-book book-name attach zone trust
user@host# set security address-book book-name attach zone dot1x
```

2. Configure security source rule-set from trust zone to the WPA authentication.

```
[edit]
user@host# set security nat source rule-set rule-set-name from zone trust
user@host# set security nat source rule-set   rule-set-name to zone dot1x
```

3. Configure the security source to match the source and destination address.

```
[edit]
user@host# set security nat source rule-set  rule-set-name rule rule-name match source-
address ip-address
user@host# set security nat source rule-set rule-set-name rule rule-name match destination-
address ip-address
```

4. Configure the UDP protocol and security source on the interface.

```
[edit]
user@host# set security nat source rule-set rule-set-name rule rule-name match protocol udp
user@host# set security nat source rule-set rule-set-name rule rule-name then source-nat
interface
```

5. Assign the security policies to the source and destination address.

```
[edit]
user@host# set security policies from-zone trust to-zone dot1x policy internet-access match
source-address ip-address
user@host# set security policies from-zone trust to-zone dot1x policy internet-access match
destination-address ip-address
user@host# set security policies from-zone trust to-zone dot1x policy internet-access match
application any
user@host# set security policies from-zone trust to-zone dot1x policy internet-access then
permit
```

6. Commit the configuration.

After completing the configuration successfully completed, you can view the parameters by using the `show wlan access-points` *name* `virtual-access-points` command.

Configure Multiple VLANs and SSIDs

You can configure 8 VAPs on each radio and each VAP is identified by the SSID. Up to 16 SSIDs can be configured on the Wi-Fi Mini-PIM. You can map a VLAN to each SSID or you can assign a single VLAN for multiple SSIDs The client connects to the VAP using the SSID and is associated to the VLAN that is mapped to the SSID.

You can configure multiple SSIDs to provide varied levels of access to different devices and users. Here is a sample configuration for three different types of users connecting to different VAPs. Each VAP is associated with a different VLAN.

Interface	VLAN ID	Address pool	VAP	SSID	Address pool
wl-2/0/0.0	100	junosDHCPPool			192.168.2.0/24
wl-2/0/0.10	10	junosDHCPPool1	VAP1	VAP-10	192.168.10.0/24
wl-2/0/0.20	20	junosDHCPPool2	VAP2	VAP-20	192.168.20.0/24
wl-2/0/0.30	30	junosDHCPPool3	VAP3	VAP-30	192.168.30.0/24

1. Configure DHCP service for WL interface.

```
user@host# set interfaces wl-2/0/0 unit 0 vlan-id 100
user@host# set interfaces wl-2/0/0 unit 0 family inet address 192.168.2.1/24
```

2. Configure Rule set for the zone.

```
user@host# set security nat source rule-set rs1 from zone trust2
user@host# set security nat source rule-set rs1 to zone trust
user@host# set security nat source rule-set rs1 rule r1 match source-address 0.0.0.0/0
user@host# set security nat source rule-set rs1 rule r1 match destination-address
192.168.11.1/32
user@host# set security nat source rule-set rs1 rule r1 then source-nat interface
```

3. Configure policy and zone for the wl interface.

```
user@host# set security policies from-zone trust2 to-zone trust policy internet-access match
source-address any
user@host# set security policies from-zone trust2 to-zone trust policy internet-access match
destination-address any
user@host# set security policies from-zone trust2 to-zone trust policy internet-access match
application any
user@host# set security policies from-zone trust2 to-zone trust policy internet-access then
permit
user@host# set security policies default-policy permit-all
user@host# set security zones security-zone trust host-inbound-traffic system-services all
user@host# set security zones security-zone trust host-inbound-traffic protocols all
user@host# set security zones security-zone trust2 host-inbound-traffic system-services all
user@host# set security zones security-zone trust2 host-inbound-traffic protocols all
user@host# set security zones security-zone trust2 interfaces wl-3/0/0.2
user@host# set security zones security-zone trust2 interfaces wl-3/0/0.1
```

4. Configure interface with VLAN tagging for multiple VLANs.

```
user@host# set interfaces wl-3/0/0 vlan-tagging
user@host# set interfaces wl-3/0/0 unit 1 vlan-id 11
user@host# set interfaces wl-3/0/0 unit 1 family inet address 192.168.200.1/24
user@host# set interfaces wl-3/0/0 unit 2 vlan-id 10
user@host# set interfaces wl-3/0/0 unit 2 family inet address 192.168.10.1/24
```

5. Configure DHCP address Assignment for the AP pools.

```
user@host# set access address-assignment pool p1 family inet network 192.168.10.0/24
user@host# set access address-assignment pool p1 family inet range r1 low 192.168.10.11
user@host# set access address-assignment pool p1 family inet range r1 high 192.168.10.128
user@host# set access address-assignment pool p1 family inet dhcp-attributes router
192.168.10.0
user@host# set access address-assignment pool p2 family inet network 192.168.200.0/24
user@host# set access address-assignment pool p2 family inet range r1 low 192.168.200.11
user@host# set access address-assignment pool p2 family inet range r1 high 192.168.200.128
user@host# set access address-assignment pool p2 family inet dhcp-attributes router
192.168.200.0
user@host# set wlan access-point name radio 1 virtual-access-point 1 maximum-stations 70
```

6. Configure Access Points & multiple SSIDs for both the raidos.

```
user@host# set wlan access-point name interface wl-3/0/0
user@host# set wlan access-point name radio 1 radio-options channel number auto
user@host# set wlan access-point name radio 1 virtual-access-point 0 ssid VAP-50
user@host# set wlan access-point name radio 1 virtual-access-point 0 vlan 11
user@host# set wlan access-point name radio 1 virtual-access-point 0 security wpa-enterprise
wpa-version v2
user@host# set wlan access-point name radio 1 virtual-access-point 0 security wpa-enterprise
cipher-suites ccmp
user@host# set wlan access-point name radio 1 virtual-access-point 0 security wpa-enterprise
radius-server 192.168.11.1
user@host# set wlan access-point name radio 1 virtual-access-point 0 security wpa-enterprise
radius-port 1823
user@host# set wlan access-point name radio 1 virtual-access-point 0 security wpa-enterprise
radius-key "$9$N7-YoDjqfQnk.nCpBSy8X7-s2oJGiqm"
user@host# set wlan access-point name radio 2 radio-options channel number auto
user@host# set wlan access-point name radio 2 virtual-access-point 0 ssid VAP-20
user@host# set wlan access-point name radio 2 virtual-access-point 0 vlan 10
user@host# set wlan access-point name radio 2 virtual-access-point 0 security wpa-enterprise
wpa-version v2
user@host# set wlan access-point name radio 2 virtual-access-point 0 security wpa-enterprise
cipher-suites ccmp
user@host# set wlan access-point name radio 2 virtual-access-point 0 security wpa-enterprise
radius-server 192.168.11.1
user@host# set wlan access-point name radio 2 virtual-access-point 0 security wpa-enterprise
radius-port 1823
```

```
user@host# set wlan access-point name radio 2 virtual-access-point 0 security wpa-enterprise
radius-key "$9$yIRrWxbwgJUH24Hm5FAtRhSrM8xNdsgo"
```

Verification

Display information about the parameters configured on the Wi-Fi Mini-PIM.

- To verify that the access point is up:

```
user@host# show wlan access-points

Active access points information

Access-Point                          Type   Interface    Radio-mode/Channel/Bandwidth
name                                  Int    wl-3/0/0     acn/100/40, gn/1/20

user@host> show interfaces terse wl*
Interface               Admin Link Proto    Local                 Remote
wl-3/0/0                up    up
wl-3/0/0.1              up    up   inet     192.168.200.1/24
wl-3/0/0.2              up    up   inet     192.168.10.1/24
wl-3/0/0.32767          up    up
```

```
user@host# show wlan access-points virtual-access-points all name

Virtual access points information

Access point name: name
Radio1:
  VAP0:
    SSID                    : VAP-50
    MAC Address             : 94:f7:ad:2c:08:41
    Maximum Station         : 127
    Broadcast SSID          : Enable
    Station Isolation       : Disable
    Upload Limit            : Disable
    Download Limit          : Disable
    VLAN ID                 : 11
    Captive Portal          : Disable
```

```
    Station MAC Filter      : Disable
    Traffic Statistics:
      Input Bytes           : 0
      Output Bytes          : 0
      Input Packets         : 0
      Output Packets        : 0
Radio2:
  VAP0:
    SSID                    : VAP-20
    MAC Address             : 94:f7:ad:2c:08:42
    Maximum Station         : 127
    Broadcast SSID          : Enable
    Station Isolation       : Disable
    Upload Limit            : Disable
    Download Limit          : Disable
    VLAN ID                 : 10
    Captive Portal          : Disable
    Station MAC Filter      : Disable
    Traffic Statistics:
      Input Bytes           : 0
      Output Bytes          : 0
      Input Packets         : 0
      Output Packets        : 0
```

- Client associations on 2.4g:

```
user@host# show wlan access-points client-associations name
Access point client associations information
Access point: name
VAP                                     Client MAC Address   Auth   Packets Rx/Tx   Bytes
Rx/Tx
Radio2:VAP-20                           e2:08:11:8a:d1:f5    OK     628/3
19172/681
```

- Client associations on 5g:

```
user@host# show wlan access-points client-associations name
Access point client associations information
Access point: name
VAP                                     Client MAC Address   Auth   Packets Rx/Tx   Bytes
```

```
Rx/Tx
Radio1:VAP-50                          b6:8a:c4:bf:08:74    OK      72/4
4302/1032
```

RELATED DOCUMENTATION

wlan

Platform-Specific Wi-Fi Mini-Physical Interface Support Behavior

Use Feature Explorer to confirm platform and release support for specific features.

Use the following table to review platform-specific behaviors for your platform:

Platform	Difference
SRX Series	SRX320, SRX340, SRX345, SRX380, and SRX550M devices that support Wi-Fi Mini-Physical Interface Module (Wi-Fi Mini-PIM) provide an integrated wireless access point —or wireless LAN— along with routing, switching, and security in a single device.

CHAPTER 9

Supported Interfaces for Security Devices

IN THIS CHAPTER

Configuring 1-Port Clear Channel DS3/E3 GPIM

IN THIS SECTION

The 1-Port Clear Channel DS3/E3 GPIM is a channel interface that can support full-duplex DS3 (T3) or E3 line rates. The below topics shows the overview of the interface, examplse on how to configure the 1-Port Clear Channel DS3/E3 GPIM for DS3 port mode, E3 port mode and M23 mapping mode respectively.

Understanding the 1-Port Clear Channel DS3/E3 GPIM

IN THIS SECTION

The 1-Port Clear Channel DS3/E3 Gigabit-Backplane *Physical Interface Module* (GPIM) for the device functions as a clear channel interface that can support full-duplex DS3 (T3) or E3 line rates of 44.796 or 34.368 Mbps, respectively. The DS3/E3 interface is a popular high-bandwidth WAN interface for large enterprise branch locations that enables high-quality voice, video, and data applications with reduced latency. The GPIM device does not support channelization, but it supports a subrate DS3/E3 configuration.

This topic includes the following sections:

Supported Features

The clear channel implementation provides such features as subrate and scrambling options used by major DSU vendors. The following key features are available depending on the interface and mode selections:

- Framed and unframed DS3 (default) and E3 port modes
- Support for frame relay, point-to-point, and HDLC serial encapsulation protocols
- Support for popular vendor algorithms for subrate and payload scrambling
- Support for generation and detection of loopback control codes (line-loopback activate and deactivate) and FEAC codes
- External and internal clocking support
- Support for DS3 and E3 network alarms
- Support for chassis clusters
- Support for anti-counterfeit check
- Loopback (local, remote, and payload) and BERT/PRBS/QRSS diagnostics support
- MTU size of 4474 bytes (default) and 9192 bytes (maximum)

Interface Naming

The following format represents the 1-Port Clear Channel DS3/E3 GPIM interface names:

```
type-fpc/pic/port
```

where:

- *type*—Media type (T3 or E3)
- *fpc*—Number of the Flexible PIC Concentrator (FPC) card on which the physical interface is located
- *pic*—Number of the PIC on which the physical interface is located
- *port*—Specific port on the PIC

Examples: t3-1/0/0 and e3-2/0/0

Physical Interface Settings

The 1-Port Clear Channel DS3/E3 GPIM supports IP configurations. Using the CLI, you can configure the 1-Port Clear Channel DS3/E3 GPIM to operate in either DS3 or E3 mode. By default, at installation the physical interface, t3-x/y/z, is enabled on the GPIM port operating in DS3 mode with T3 framing.

You can reset the mode of the physical interface to E3 using the edit chassis command:

```
[edit]
user@host# set chassis fpc 1 pic 0 port 0 framing e3
```

Logical Interface Settings

The *logical interface* for the device is determined by setting the t3-options or e3-options of the edit interfaces command.

You can specify the MTU size for the GPIM interface. Junos OS supports an MTU value of 4474 bytes for the default value or up to 9192 bytes for maximum jumbo GPIM implementations.

Table 56 on page 448 identifies network interface specifications for DS3 or E3 modes.

Table 56: 1-Port Clear Channel DS3/E3 GPIM Interface Options

Description	DS3 Mode	E3 Mode
Network Interface Specifications		
Line encoding	B3ZS	HDB3
Framing	• C-bit parity (default) • M23	G.751 (default)

Table 56: 1-Port Clear Channel DS3/E3 GPIM Interface Options *(Continued)*

Description	DS3 Mode	E3 Mode
Subrate and scrambling	Vendor algorithms supported: • Adtran • Digital Link • Kentrox • Larscom • Verilink	Vendor algorithms supported: • Digital Link • Kentrox
Network alarms	Supported in accordance with the ANSI specification: • Loss of signal (LOS) • Out of frame (OOF) • Loss of frame (LOF) • Alarm identification Signal (AIS) • Remote defect identification (RDI)	Supported in accordance with the ITU-T specification: • Loss of signal (LOS) • Out of frame (OOF) • Alarm identification signal (AIS) • Remote defect identification (RDI) • Phase- locked loop (PLL)

Table 56: 1-Port Clear Channel DS3/E3 GPIM Interface Options *(Continued)*

Description	DS3 Mode	E3 Mode
Error counters	Incremented during a periodic 1-second polling routine: • Line code violations (LCV) • P-bit code violations (PCV) • C-bit code violations (CCV) • Line errored seconds (LES) • P-bit errored seconds (PES) • C-bit errored seconds (CES) • Severely errored framing seconds (SEFS) • P-bit severely errored seconds (PSES) • C-bit severely errored seconds (CSES) • Unavailable seconds (UAS)	Incremented during a periodic 1-second polling routine: • Frame alignment error (FAE) • Bipolar coding violations (BCV) • Excessive zeros (EXZ) • Line code violations (LCV) • Line errored seconds (LES) • Severely errored framing seconds (SEFS) • Unavailable seconds (UAS)
HDLC Features		
MTU	Default (4474 bytes) or maximum jumbo (up to 9192 bytes)	Default (4474 bytes) or maximum jumbo (up to 9192 bytes)
Shared flag	Supported	Supported
Idle flag/fill (0x7e or all ones)	Supported	Supported
Counters	Runts, giants	Runts, giants

SEE ALSO

Interface Naming Conventions | 8

Example: Configuring the 1-Port Clear-Channel DS3/E3 GPIM for DS3 Port Mode

IN THIS SECTION

This example configures the GPIM in the DS3 (T3) operation mode.

Requirements

Before you begin:

- Install the device as specified in the *SRX Series Services Physical Interface Modules Hardware Guide*.

Overview

IN THIS SECTION

This example configures the basic T3 interface and modifies the framing to C-bit parity mode.

Topology

Configuration

IN THIS SECTION

- Procedure | 452

Procedure

Step-by-Step Procedure

To configure the GPIM:

1. Verify the installation, location, and status of the GPIM. In this example, the GPIM is installed in slot 8/PIC 0 and is currently online.

```
user@host> show chassis fpc pic-status
```

```
Slot 0 Online FPC
   PIC 0 Online 4x GE Base PIC
Slot 2 Offline FPC
Slot 5 Offline FPC
Slot 6 Online FPC
   PIC 0 Online 4x CT1E1 gPIM
Slot 7 Offline FPC
Slot 8 Online FPC
   PIC 0 Online 1x CLR CH T3/E3
```

2. Set the IP address for the logical interface.

```
[edit]
user@host# set interfaces t3-8/0/0 unit 0 family inet address interface 192.107.1.230/24
```

3. Set the MTU value to 9018.

```
[edit]
user@host# set interfaces t3-8/0/0 unit 0 family inet mtu 9018
```

4. Set the framing mode.

```
[edit]
user@host# set interfaces t3-8/0/0 t3-options cbit-parity
```

5. Enable the unframed DS3 mode.

```
[edit]
user@host# set interfaces t3-8/0/0 t3-options unframed
```

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

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

7. To verify the configuration for your device, enter the following operational command:

```
user@host> show interfaces t3-8/0/0 extensive
```

Example: Configuring the 1-Port Clear Channel DS3/E3 GPIM for E3 Port Mode

IN THIS SECTION

This example modifies the default configuration for an E3 environment.

Requirements

Before you begin:

- Install the device as specified in the *SRX Series Services Physical Interface Modules Hardware Guide*.

Overview

IN THIS SECTION

This example configures the basic E3 interface.

Topology

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To configure the GPIM in E3 framing:

1. Verify the installation, location, and status of the GPIM. In this example, the GPIM is installed in slot 8/PIC 0 and is currently online.

```
user@host> show chassis fpc pic-status
```

```
Slot 0 Online FPC
   PIC 0 Online 4x GE Base PIC
Slot 2 Offline FPC
Slot 5 Offline FPC
Slot 6 Online FPC
   PIC 0 Online 4x CT1E1 gPIM
Slot 7 Offline FPC
Slot 8 Online FPC
   PIC 0 Online 1x CLR CH T3/E3
```

2. Change to E3 port mode.

```
[edit]
user@host# set chassis fpc 8 pic 0 port 0 framing e3
```

3. Reset the MTU value to 3474.

```
[edit]
user@host# set interfaces e3-8/0/0 unit 0 family inet mtu 3474
```

4. Enable the unframed mode.

```
[edit]
user@host# set interfaces e3-8/0/0 e3-options unframed
```

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

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

6. To verify the configuration for your device, enter the following operational command:

```
user@host> show interfaces e3-8/0/0 extensive
```

Example: Configuring the 1-Port Clear-Channel DS3/E3 GPIM for M23 Mapping Mode

IN THIS SECTION

The following example configures the GPIM in DS3 with M23 mapping mode. Note that M23 mapping does not provide C-bit parity.

Requirements

Before you begin:

- Install the device as specified in the *SRX Series Services Physical Interface Modules Hardware Guide*.

Overview

IN THIS SECTION

This example configures the basic T3 interface and modifies the framing to M23 mode without C-bit parity.

Topology

Configuration

IN THIS SECTION

- Procedure | 457

Procedure

Step-by-Step Procedure

To configure the GPIM:

1. Verify the installation, location, and status of the GPIM. In this example, the GPIM is installed in slot 8/PIC 0 and is currently online.

```
user@host> show chassis fpc pic-status
```

```
Slot 0 Online FPC
   PIC 0 Online 4x GE Base PIC
Slot 2 Offline FPC
Slot 5 Offline FPC
Slot 6 Online FPC
   PIC 0 Online 4x CT1E1 gPIM
Slot 7 Offline FPC
Slot 8 Online FPC
   PIC 0 Online 1x CLR CH T3/E3
```

2. Set the IP address for the logical interface.

```
[edit]
user@host# set interfaces t3-8/0/0 unit 0 family inet address interface 192.107.1.230/24
```

3. Set the MTU value to 9018.

```
[edit]
user@host# set interfaces t3-8/0/0 unit 0 family inet mtu 9018
```

4. Set the framing mode.

```
[edit]
user@host# set interfaces t3-8/0/0 t3-options m23
```

5. Disable C-bit parity for M23 mode.

```
[edit]
user@host# set interfaces t3-8/0/0 t3-options no-cbit-parity
```

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

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

7. To verify the configuration for your device, enter the following operational command:

```
user@host> show interfaces t3-8/0/0 extensive
```

Configuring 3G Wireless Modems for WAN Connections

IN THIS SECTION

The topics below discuss the overview and configuration of 3G Wireless Modem, dialer interface, and 3G Wireless Modem physical interface.

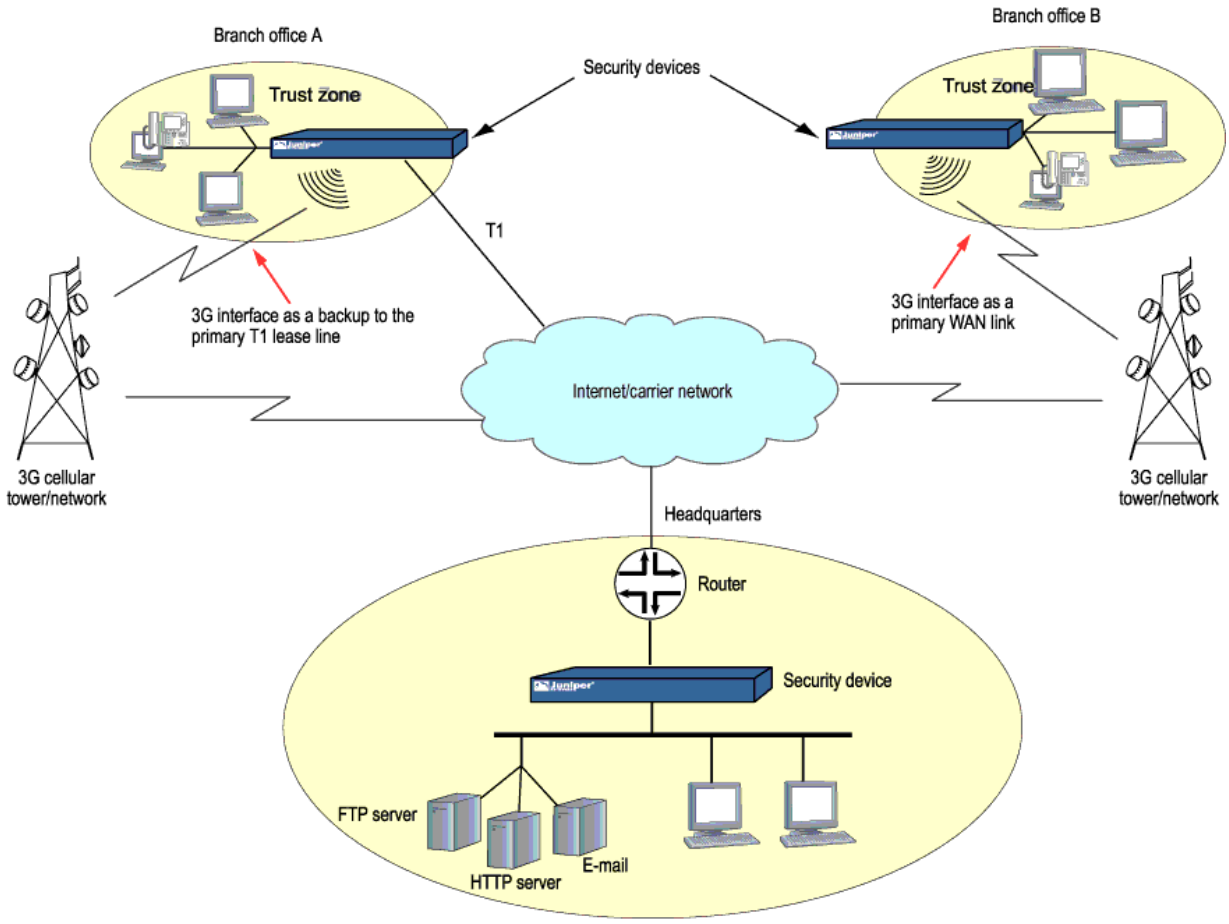
3G Wireless Modem Overview

3G refers to the third generation of mobile phone standards and technology based on the International Telecommunication Union (ITU) International Mobile Telecommunications-2000 (IMT-2000) global standard. 3G networks are wide area cellular telephone networks that have evolved to include high-data rate services of up to 3 Mbps. This increased bandwidth makes 3G networks a viable option as primary or backup wide area network (WAN) links for a branch office.

Juniper Networks security devices support 3G wireless interfaces (USB-based 3G modems). When used in a branch office, these devices can provide dial-out services to PC users and forward IP traffic through a service provider's cellular network.

Figure 24 on page 460 illustrates a basic setup for 3G wireless connectivity for two branch offices. Branch Office A has a T1 leased line as the primary wide area network (WAN) link and a 3G wireless modem connection as the failover link. Branch Office B uses the 3G wireless modem connection as the primary WAN link.

Figure 24: Wireless WAN Connections for Branch Offices



3G Wireless Modem Configuration Overview

Before you begin:

1. Install your SRX Series Firewall and establish basic connectivity for your device. For more information, see the SRX Series Hardware Guide for your device.
2. Obtain a supported 3G wireless modem card for the device.
3. Establish an account with a cellular network service provider. Contact your service provider for more information.
4. With the services gateway powered off, insert the 3G wireless modem card into the ExpressCard slot (SRX320 devices) or 3G USB modems (SRX300 devices). Power on the device. The EXPCARD LED (for SRX320) and 3G LED (SRX320) on the front panel of the device indicates the status of the 3G wireless modem interface.

WARNING: The device must be powered off before you insert the 3G wireless modem card in the ExpressCard slot (SRX320) or integrated 3G USB modem (SRX320). Do not insert or remove the card when the device is powered on.

To configure and activate the 3G wireless modem card:

1. Configure a dialer interface. See "Example: Configuring the Dialer Interface" on page 464.
2. Configure the 3G wireless modem interface. See "Example: Configuring the 3G Wireless Modem Interface" on page 473.
3. Configure security zones and policies, as needed, to allow traffic through the WAN link. See *Example: Creating Security Zones*.

To use the 3G USB modems on the SRX210 device:

1. Upgrade the BIOS software packaged inside the Junos OS image. For detailed information about BIOS upgrade procedures, see the Software Installation and Upgrade Guide.

   NOTE: You need the BIOS version of 2.1 or higher to use the 3G USB modems on the SRX210 device.

2. Configure the WAN port using the CLI command `set chassis routing-engine usb-wwan port 1` to enable the USB port to use the U319 USB modem.
3. Plug the 3G USB modem in to the appropriate USB slot (USB port 1) on the device.

   NOTE: You can use the USB modem with a standard USB extension cable of 1.8288 meters (6 ft) or longer.

4. Reboot the device to start using the 3G USB modem.

Understanding the Dialer Interface

IN THIS SECTION

The *dialer interface*, `dln`, is a *logical interface* for configuring properties for modem connections. You can configure multiple dialer interfaces on an SRX Series Firewall. A dialer interface and a dialer pool (which includes the physical interface) are bound together in a dialer profile.

The dialer interface for 3G wireless modems is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

This topic contains the following sections:

Dialer Interface Configuration Rules

The following rules apply when you configure dialer interfaces for 3G wireless modem connections:

- The dialer interface must be configured to use the default Point-to-Point Protocol (PPP) encapsulation. You cannot configure Cisco High-Level Data Link Control (HDLC) or Multilink PPP (MLPPP) encapsulation on dialer interfaces.
- You cannot configure the dialer interface as a constituent link in a multilink bundle.
- You cannot configure any dial-in options for the dialer interface.

You configure the following for a dialer interface:

- A dialer pool to which the physical interface belongs.
- Source IP address for the dialer interface.
- Dial string (optional) is the destination number to be dialed.
- Authentication, for GSM HSDPA 3G wireless modem cards.
- Watch list, if the dialer interface is a backup WAN link.

With GSM HSDPA 3G wireless modem cards, you might need to configure PAP or CHAP for authentication with the service provider network. The service provider must supply the username and password, which you configure in an access profile. You then specify the access profile in a dialer interface.

Next you set the dialer interface as a backup WAN link to a primary interface. Then you create a dialer watch to enable the device to monitor the route to a head office router and set a dialer pool. Finally, you create a dialer filter firewall rule for traffic from the branch office to the main office router and associate the dialer filter with a dialer interface.

Dialer Interface Authentication Support for GSM HSDPA 3G Wireless Modems

For GSM HSDPA 3G wireless modems, you configure a dialer interface to support authentication through Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP).

CHAP is a server-driven, three-step authentication method that depends on a shared secret password that resides on both the server and the client. When you enable CHAP on a dialer interface, the device can authenticate its peer and be authenticated by its peer.

PAP allows a simple method for a peer to establish its identity using a two-way handshake during initial link establishment. After the link is established, an identification and password pair is repeatedly sent by the peer to the authenticator until authentication is acknowledged or the connection is terminated.

Dialer Interface Functions

The dialer interface can perform backup, dialer filter, and dialer watch functions, but these operations are mutually exclusive. You can configure a single dialer interface to operate in only one of the following ways:

- As a backup interface for a single primary WAN connection. The dialer interfaces are activated only when the primary interface fails. The 3G wireless modem backup connectivity is supported on all interfaces except `lsq-0/0/0`.
- As a dialer filter. The Dialer filter enables the 3G wireless modem connection to be activated only when specific network traffic is sent on the backup WAN link. You configure a firewall rule with the dialer filter option, and then apply the dialer filter to the dialer interface.
- As a dialer watch interface. With dialer watch, the SRX Series Firewall monitors the status of a specified route and if the route disappears, the dialer interface initiates the 3G wireless modem connection as a backup connection. To configure dialer watch, you first add the routes to be monitored to a watch list in a dialer interface; specify a dialer pool for this configuration. Then configure the 3G wireless modem interface to use the dialer pool.

Dialer Interface Operating Parameters

You can also specify optional operating parameters for the dialer interface:

- Activation delay—Number of seconds after the primary interface is down before the backup interface is activated. The default value is 0 seconds, and the maximum value is 60 seconds. Use this option only if dialer watch is configured.
- Deactivation delay—Number of seconds after the primary interface is up before the backup interface is deactivated. The default value is 0 seconds, and the maximum value is 60 seconds. Use this option only if dialer watch is configured.
- Idle timeout—Number of seconds the connection remains idle before disconnecting. The default value is 120 seconds, and the range is from 0 to 4,294,967,295 seconds.
- Initial route check—Number of seconds before the primary interface is checked to see if it is up. The default value is 120 seconds, and the range is from 1 to 300 seconds.

Example: Configuring the Dialer Interface

IN THIS SECTION

This example shows how to configure the dialer interface for 3G wireless modem connections.

The dialer interface for 3G wireless modems is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Requirements

Before you begin, install your SRX Series Firewall and establish basic connectivity for your device. See "3G Wireless Modem Configuration Overview" on page 460.

Overview

In this example, you first configure the dialer interface as dl0, specify the PPP encapsulation dialer pool as 1, specify the dial string as 14691, and negotiate the address option for the interface IP address.

Configuration

IN THIS SECTION

Configuring a Dialer Interface

CLI Quick Configuration

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

```
set interfaces dl0 description 3g-wireless encapsulation ppp unit 0 dialer-options pool 1 dial-
string 14691
set interfaces dl0 unit 0 family inet negotiate-address
```

Step-by-Step Procedure

1. Set the interface and specify the PPP encapsulation, dialer pool, and dial string.

```
[edit]
user@host# set interfaces dl0 description 3g-wireless encapsulation ppp unit 0 dialer-options
pool 1 dial-string 14691
```

2. Set the negotiate address option for the interface IP address.

```
[edit]
user@host# set interfaces dl0 unit 0 family inet negotiate-address
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces dl0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interfaces dl0
description 3g-wireless;
encapsulation ppp;
    unit 0 {
family inet {
negotiate-address;
     }
dialer-options {
pool 1;
    dial-string 14691;
     }
}
```

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

Configuring PAP on the Dialer Interface

CLI Quick Configuration

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

```
set access profile pap-1 client clientX pap-password 7a^6b%5c
set interfaces dl0 unit 0 ppp-options pap access-profile pap-1
```

Step-by-Step Procedure

1. Configure a PAP access profile.

```
[edit]
user@host# set access profile pap-1 client clientX pap-password 7a^6b%5c
```

2. Associate the PAP access profile with a dialer interface.

```
[edit]
user@host# set interfaces dl0 unit 0 ppp-options pap access-profile pap-1
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces dl0` and `show access profile pap-1` commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interfaces dl0
unit 0 {
ppp-options {
    pap {
access-profile pap-1;
}
     }
}
[edit]
    user@host# show access profile pap-1
client clientX pap-password "$9$jnqTz3nCBESu01hSrKvZUDkqf"; ## SECRET-DATA
```

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

Configuring CHAP on the Dialer Interface

CLI Quick Configuration

With GSM HSDPA 3G wireless modem cards, you may need to configure CHAP for authentication with the service provider network. The service provider must supply the username and password, which you configure in an access profile. You then specify this access profile in a dialer interface.

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

```
set access profile chap-1 client clientX chap-secret 7a^6b%5c
set interfaces dl0 unit 0 ppp-options chap access-profile chap-1
```

Step-by-Step Procedure

1. Configure a CHAP access profile.

```
[edit]
user@host# set access profile chap-1 client clientX chap-secret 7a^6b%5c
```

2. Associate the CHAP access profile with a dialer interface.

```
[edit]
user@host# set interfaces dl0 unit 0 ppp-options chap access-profile chap-1
```

Results

From configuration mode, confirm your configuration by entering the `show access profile chap-1` and `show interfaces dl0` commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show access profile chap-1
client clientX chap-secret "$9$neYpCO1REyWx-Kv87-VsYQF39Cu"; ## SECRET-DATA
[edit]
    user@host# show interfaces dl0
```

```
unit 0 {
ppp-options {
    chap {
access-profile chap-1;
}
    }
}
```

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

Configuring the Dialer Interface as a Backup WAN Connection

CLI Quick Configuration

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

```
set interfaces ge-0/0/1 unit 0 backup-options interface dl0
```

Step-by-Step Procedure

1. Set interface back up option.

```
[edit]
user@host# set interfaces ge-0/0/1 unit 0 backup-options interface dl0
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces ge-0/0/1` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interfaces ge-0/0/1
unit 0 {
backup-options {
    interface dl0.0;
```

```
    }
}
```

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

Configuring Dialer Watch for the 3G Wireless Modem Interface

CLI Quick Configuration

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

```
set interfaces dl0 description dialer-watch unit 0 dialer-options watch-list 200.200.201.1/32
set interfaces dl0 description dialer-watch unit 0 dialer-options pool dw-pool
```

Step-by-Step Procedure

1. Create a dialer watch.

```
[edit]
user@host# set interfaces dl0 description dialer-watch unit 0 dialer-options watch-list
200.200.201.1/32
```

2. Set a dialer pool.

```
[edit]
user@host# set interfaces dl0 description dialer-watch unit 0 dialer-options pool dw-pool
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces dl0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interfaces dl0
description dialer-watch;
```

```
unit 0 {
    dialer-options {
watch-list {
200.200.201.1/32;
    }
    }
}
```

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

Configuring a Dialer Filter for the 3G Wireless Modem Interface

CLI Quick Configuration

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

```
set firewall family inet dialer-filter traffic-filter term term1 then note
```

Step-by-Step Procedure

1. Associate the dialer filter with a dialer interface.

```
[edit]
user@host# set firewall family inet dialer-filter traffic-filter term term1 then note
```

2. Check your other changes to the configuration before committing.

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

Results

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

```
[edit]
user@host# show firewall
family inet {
    dialer-filter traffic-filter {
        term term-1 {
            then note;
        }
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Configuration

Purpose

Verify the configuration output.

Action

Verify the configuration output by entering the `show interfaces` command.

Understanding the 3G Wireless Modem Physical Interface

You configure two types of interfaces for 3G wireless modem connectivity—the physical interface and a logical dialer interface.

The physical interface for the 3G wireless modem uses the name `cl-0/0/8`. This interface is automatically created when a 3G wireless modem is installed in the device.

The 3G wireless modem physical interface is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

You configure the following properties for the physical interface:

- A dialer pool to which the physical interface belongs and the priority of the interface in the pool. A physical interface can belong to more than one dialer pool. The dialer pool priority has a range from `1` to `255`, with `1` designating the lowest-priority interfaces and `255` designating the highest-priority interfaces.
- Modem initialization string (optional). These strings begin with `AT` and execute Hayes modem commands that specify modem operation.
- GSM profile for establishing a data call with a GSM cellular network.

By default, the modem allows access to networks other than the home network.

Example: Configuring the 3G Wireless Modem Interface

IN THIS SECTION

This example shows how to configure the 3G wireless modem interface.

The 3G wireless modem physical interface is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Requirements

Before you begin, configure a dialer interface. See "Example: Configuring the Dialer Interface" on page 464.

Overview

In this example, you configure the physical interface as cl-0/0/8 for the 3G wireless modem to use dialer pool 1 and set the priority for the dialer pool to 25. You also configure a modem initialization string to autoanswer after two rings.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To configure the 3G wireless modem interface:

1. Specify the dialer pool.

```
[edit]
user@host# set interfaces cl-0/0/8 dialer-options pool 1 priority 25
```

2. Specify the modem options.

```
[edit]
user@host# set interfaces cl-0/0/8 modem-options init-command-string “ATSO=2\n”
```

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

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

Verification

To verify the configuration is working properly, enter the `show interfaces cl-0/0/8 modem options` command.

Understanding the GSM Profile

To allow data calls to a Global System for Mobile Communications (GSM) network, you must obtain the following information from your service provider:

- Username and password
- Access point name (APN)
- Whether the authentication is Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP)

You configure this information in a GSM profile associated with the 3G wireless modem physical interface. You can configure up to 16 different GSM profiles, although only one profile can be active at a time.

NOTE: You also need to configure a CHAP or PAP profile with the specified username and password for the dialer interface.

Subscriber information is written to the Subscriber Identity Module (SIM) on the GSM HSDPA 3G wireless modem card. If the SIM is locked, you must unlock it before activation by using the master subsidy lock (MSL) value given by the service provider when you purchase the cellular network service.

Some service providers may preload subscriber profile information on a SIM card. The assigned subscriber information is stored in profile 1, while profile 0 is a default profile created during manufacturing. If this is the case, specify profile 1 for the GSM profile associated with the 3G wireless modem physical interface.

Configuring the information in a GSM profile associated with the 3G wireless modem physical interface is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Example: Configuring the GSM Profile

IN THIS SECTION

This example shows how to configure the GSM profile for the 3G wireless modem interface with service provider networks such as AT&T and T-Mobile.

NOTE: Configuring the information in a GSM profile associated with the 3G wireless modem physical interface is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Requirements

Before you begin:

- Configure a dialer interface. See "Example: Configuring the Dialer Interface" on page 464
- Configure the 3G wireless modem interface. See "Example: Configuring the 3G Wireless Modem Interface" on page 473.

Overview

IN THIS SECTION

In this example, you configure the following information provided by a service provider in a GSM profile called juniper99 that is associated with the 3G wireless modem physical interface cl-0/0/8:

- Username—`juniper99`

- Password—1@#6ahgfh
- Access point name (APN)—apn.service.com
- Authentication method—CHAP

Then you activate the profile by specifying the profile ID as profile-id 1.

Topology

Configuration

IN THIS SECTION

- Procedure | 477

Procedure

Step-by-Step Procedure

To configure a GSM profile for the 3G wireless modem interface:

1. Create a GSM profile.

```
[edit]
user@host> request modem wireless gsm create-profile profile-id 1 sip-user-id juniper99 sip-
password 16ahgfh access-point-name apn.service.com authentication-method chap
```

2. Activate the profile.

```
[edit]
user@host# set interface cl-0/0/8 cellular-options gsm-options select-profile profile-id 1
```

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

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

Verification

To verify the configuration is working properly, enter the `show interfaces cl-0/0/8` command.

Unlocking the GSM 3G Wireless Modem

Before you begin, obtain the PIN from the service provider.

The subscriber identity module (SIM) in the GSM 3G wireless modem card is a detachable smart card. Swapping out the SIM allows you to change the service provider network, however some service providers lock the SIM to prevent unauthorized access to the service provider's network. If this is the case, you will need to unlock the SIM by using an personal identification number (PIN), a four-digit number provided by the service provider.

NOTE: Unlocking the SIM in a 3G wireless modem card is not supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Use the CLI operational mode command to unlock the SIM on the GSM 3G wireless modem card.

This example uses the PIN `3210` from the service provider.

To unlock the SIM on the GSM 3G wireless modem card:

```
user@host> request modem wireless gsm sim-unlock cl-0/0/8 pin 3210
```

A SIM is blocked after three consecutive failed unlock attempts; this is a security feature to prevent brute force attempts to unlock the SIM. When the SIM is blocked, you need to unblock the SIM with an eight-digit PIN unlocking key (PUK) obtained from the service provider.

To unlock the SIM automatically on reboot:

```
user@host# set interfaces cl-0/0/8 cellular-options gsm-options sim-unlock-code
Enter PIN:
user@host#
```

NOTE: On SRX300, SRX320 devices, when you power on or reboot the device, the Subscriber Identity Module (SIM) will be locked. If the SIM Personal Identification Number (PIN) or the unlock code is configured in the `set interfaces cl-0/0/8 cellular-`

options gsm-options sim-unlock-code configuration command, then Junos OS attempts to unlock the SIM only once. This is to keep the SIM from being blocked. If the SIM is blocked, you must provide a PIN Unblocking Key (PUK) obtained from the service provider. If the wrong SIM PIN is configured, the SIM will remain locked, and the administrator can unlock it by using the remaining two attempts.

Use the CLI operational mode command to unblock the SIM.

This example uses the PUK 76543210 from the service provider.

To unblock the SIM:

```
user@host> request modem wireless gsm sim-unblock cl-0/0/8 puk 76543210
```

NOTE: If you enter the PUK incorrectly ten times, you will need to return the SIM to the service provider for reactivation.

Configuring CDMA EV-DO Modem Cards

IN THIS SECTION

The below topics discuss the account activation for CDMA EV-DO Modem Cards and activation details on security devices.

Understanding Account Activation for CDMA EV-DO Modem Cards

IN THIS SECTION

Account activation is the process of enabling the CDMA EV-DO wireless modem card to connect to your service provider's cellular network. This is a one-time process where your subscriber information is saved in nonvolatile memory on the card. The procedure you use to perform account activation depends upon the service provider network.

NOTE: Activating an account for a CDMA EV-DO 3G wireless modem card is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Before activating an account, you can verify the signal strength on the 3G wireless modem interface by using the `show modem wireless interface cl-0/0/8 rssi` command. The signal strength should be at least -90 dB and preferably better than -80 dB (-125 dB indicates nil signal strength). If the signal strength is below -90 dB, activation may not be possible from that location. For example:

```
user@host> show modem wireless interface cl-0/0/8 rssi
Current Radio Signal Strength (RSSI) = -98 dBm
```

This topic contains the following sections:

Obtaining Electronic Serial Number (ESN)

The service provider requires the electronic serial number (ESN) of the 3G wireless modem card to activate your account and to generate the necessary information you need to activate the card. You can obtain the ESN number of the modem card in the following ways:

- Inspect the modem card itself; the ESN is printed on the card.

- Use the CLI `show modem wireless interface cl-0/0/8 firmware` command, as shown in the following example, and note the value for the Electronic Serial Number (ESN) field:

```
user@host> show modem wireless interface cl-0/0/8 firmware
Modem Firmware Version : p2005600
Modem Firmware built date : 12-09-07
Card type : Aircard 597E - CDMA EV-DO revA
Manufacturer : Sierra Wireless, Inc.
Hardware Version : 1.0
Electronic Serial Number (ESN) : 0x6032688F
Preferred Roaming List (PRL) Version : 20224
Supported Mode : 1xev-do rev-a, 1x
Current Modem Temperature : 32 degrees Celsius
Modem Activated : YES
Activation Date: 2-06-08
Modem PIN Security : Unlocked
Power-up lock : Disabled
```

Account Activation Modes

For the CDMA EV-DO 3G wireless modem card, account activation can be done through one or more of the following modes:

- Over the air service provisioning (OTASP)—protocol for programming phones over the air using Interim Standard 95 (IS-95) Data Burst Messages.

  To activate the 3G wireless modem card with OTASP, you need to obtain from the service provider the dial number that the modem will use to contact the network. Typically, OTASP dial numbers begin with the feature code *228 to indicate an activation call type to the cellular network's base transceiver station, followed by additional digits specified by the service provider.

- Internet-based over the air (IOTA) provisioning—method for programming phones for voice and data services

- Manually providing the required information by entering in a CLI *operational mode command*

Sprint uses manual and IOTA activation, whereas Verizon uses only OTASP.

NOTE: The 3G wireless modem is set into Single-Carrier Radio Transmission Technology (1xRTT) mode automatically when it is activated for Verizon networks.

Activating the CDMA EV-DO Modem Card Manually

Before you begin, the service provider must activate your account before you can activate the CDMA EV-DO 3G wireless modem card.

Manual activation stores the supplied values into the 3G wireless modem card's nonvolatile memory. This topic describes the activation of the CDMA EV-DO 3G wireless modem card for use with service provider networks such as Sprint.

NOTE: Activating a CDMA EV-DO 3G wireless modem card manually is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Using the electronic serial number (ESN) you provided and your account information, the service provider supplies you with the following information for manual activation of the 3G wireless modem card:

- Master subsidy lock (MSL)—activation code
- Mobile directory number (MDN)—10-digit user phone number
- International mobile station identify (IMSI)—Mobile subscriber information
- Simple IP user identification (SIP-ID)—Username
- Simple IP password (SIP-Password)—Password

You also need to obtain the following information from the 3G wireless modem card itself for the activation:

- System identification (SID)—Number between 0 and 32767
- Network identification (NID)—Number between 0 and 65535

Use the CLI `show modem wireless interface cl-0/0/8 network` command to display the SID and NID, as shown in the following example:

```
user@host> show modem wireless interface cl-0/0/8 network
Running Operating mode : 1xEV-DO (Rev A) and 1xRTT
Call Setup Mode : Mobile IP only
System Identifier (SID) : 3421
Network Identifier (NID) : 91
Roaming Status(1xRTT) : Home
Idle Digital Mode : HDR
System Time : Wed Jun6 15:16:9 2008
```

Use the CLI operational mode command to manually activate the 3G wireless modem card.

This example uses the following values for manual activation:

- MSL (from service provider)—43210
- MDN (from service provider)—0123456789
- IMSI (from service provider)—0123456789
- SIP-ID (from service provider)—jnpr
- SIP-Password (from service provider)—jn9rl
- SID (from modem card)—12345
- NID (from modem card)—12345

To activate the CDMA EV-DO 3G wireless modem card manually:

```
user@host> request modem wireless interface cl-0/0/8 activate manual msl 43210 mdn 0123456789
imsi 0123456789 sid 12345 nid 12345 sip-id jnpr sip-password jn9rl
Checking status...
Modem current activation status: Not Activated
Starting activation...
Performing account activation step 1/6 : [Unlock] Done
Performing account activation step 2/6 : [Set MDN] Done
Performing account activation step 3/6 : [Set SIP Info] Done
Performing account activation step 4/6 : [Set IMSI] Done
Performing account activation step 5/6 : [Set SID/NID] Done
Performing account activation step 6/6 : [Commit/Lock] Done
Configuration Commit Result: PASS
Resetting the modem ... Done
Account activation in progress. It can take up to 5 minutes
Please check the trace logs for details.
```

To check the trace log for account activation details:

```
user@host> tail -f /var/log/wwand.log
Jun 25 04:42:55: IOTA cl-0/0/8 Event: IOTA Start... Success
Jun 25 04:43:45: IOTA cl-0/0/8 OTA SPL unlock...  Success
Jun 25 04:43:56: IOTA cl-0/0/8 Committing OTA Parameters to NVRAM... Success
Jun 25 04:44:02: IOTA cl-0/0/8 Over the air provisioning... Complete
Jun 25 04:44:04: IOTA cl-0/0/8 IOTA Event: IOTA End... Success
```

Activating the CDMA EV-DO Modem Card with IOTA Provisioning

Before you begin, activate the CDMA EV-DO 3G wireless modem card. See "Understanding Account Activation for CDMA EV-DO Modem Cards" on page 480.

Manual activation stores the supplied values in the 3G wireless modem card's nonvolatile memory. If the modem card is reset or you need to update Mobile IP (MIP) parameters, use the CLI operational mode command to activate the modem card with IOTA.

NOTE: Activating a CDMA EV-DO 3G wireless modem card with IOTA provisioning is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

To activate the CDMA EV-DO 3G wireless modem card with IOTA:

```
user@host> request modem wireless interface cl-0/0/8 activate iota
Beginning IOTA Activation. It can take up to 5 minutes
Please check the trace logs for details.
```

To check the trace log for account activation details:

```
user@host> tail -f /var/log/wwand.log
Jun 25 04:42:55: IOTA cl-0/0/8 Event: IOTA Start... Success
Jun 25 04:43:45: IOTA cl-0/0/8 OTA SPL unlock...  Success
Jun 25 04:43:56: IOTA cl-0/0/8 Committing OTA Parameters to NVRAM... Success
Jun 25 04:44:02: IOTA cl-0/0/8 Over the air provisioning... Complete
Jun 25 04:44:04: IOTA cl-0/0/8 IOTA Event: IOTA End... Success
```

Activating the CDMA EV-DO Modem Card with OTASP Provisioning

Before you begin:

- Obtain the dial number that the modem will use to contact the network from the service provider.
- The service provider must activate your account before OTASP provisioning can proceed.

This topic describes the activation of the CDMA EV-DO 3G wireless modem card for use with service provider networks such as Verizon.

NOTE: Activating a CDMA EV-DO 3G wireless modem card with OTASP provisioning is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Use the CLI operational mode command to activate the 3G wireless modem card.

In this example, the dial number from the service provider is *22864.

To activate the CDMA EV-DO 3G wireless modem card with OTASP provisioning:

```
user@host> request modem wireless interface cl-0/0/8 activate otasp dial-string *22864
OTASP number *22286*, Selecting NAM 0
Beginning OTASP Activation. It can take up to 5 minutes
Please check the trace logs for details.
```

To check the trace log for account activation details:

```
user@host> tail -f /var/log/wwand.log
Jun 25 04:42:55: OTASP cl-0/0/8 OTA SPL unlock... Success
Jun 25 04:43:42: OTASP cl-0/0/8 OTA PRL download... Success
Jun 25 04:43:55: OTASP cl-0/0/8 OTA Profile downloaded... Success
Jun 25 04:43:58: OTASP cl-0/0/8 OTA MDN download... Success
Jun 25 04:44:04: OTASP cl-0/0/8 Committing OTA Parameters to NVRAM... Success
Jun 25 04:44:45: Over the air provisioning... Complete
```

Configuring USB Modems for Dial Backup

IN THIS SECTION

The topics below discuss the USB modem interfaces, its configuration details, examples of configuring dialer interface, configuring PAP on dialer interface and CHAP on dialer interface.

USB Modem Interface Overview

IN THIS SECTION

Juniper Networks SRX Series Firewalls support the use of USB modems for remote management. You can use Telnet or SSH to connect to the device from a remote location through two modems over a telephone network. The USB modem is connected to the USB port on the device, and a second modem is connected to a remote management device such as a PC or laptop computer.

NOTE: USB modems are no longer supported for dial backup on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

You can configure your device to fail over to a USB modem connection when the primary Internet connection experiences interruption.

A USB modem connects to a device through modem interfaces that you configure. The device applies its own modem AT commands to initialize the attached modem. Modem setup requires that you connect and configure the USB modem at the device and the modem at the user end of the network.

You use either the J-Web configuration editor or CLI configuration editor to configure the USB modem and its supporting dialer interfaces.

NOTE: Low-latency traffic such as VoIP traffic is not supported over USB modem connections.

NOTE: We recommend using a US Robotics USB 56k V.92 Modem, model number USR Model 5637.

USB Modem Interfaces

You configure two types of interfaces for USB modem connectivity:

- A physical interface which uses the naming convention `umd0`. The device creates this interface when a USB modem is connected to the USB port.
- A *logical interface* called the dialer interface. You use the dialer interface, `dln`, to configure dialing properties for USB modem connections. The dialer interface can be configured using Point-to-Point Protocol (PPP) encapsulation. You can also configure the dialer interface to support authentication protocols—PPP Challenge Handshake (CHAP) or Password Authentication Protocol (PAP). You can configure multiple dialer interfaces for different functions on the device. After configuring the dialer interface, you must configure a backup method such as a dialer backup, a dialer filter, or a dialer watch.

The USB modem provides a dial-in remote management interface, and supports dialer interface features by sharing the same dial pool as a dialer interface. The dial pool allows the logical dialer interface and the physical interface to be bound together dynamically on a per-call basis. You can configure the USB modem to operate either as a dial-in console for management or as a dial-in WAN backup interface. Dialer pool priority has a range from 1 to 255, with 1 designating the lowest priority interfaces and 255 designating the highest priority interfaces.

Dialer Interface Rules

The following rules apply when you configure dialer interfaces for USB modem connections:

- The dialer interface must be configured to use PPP encapsulation. You cannot configure Cisco High-Level Data Link Control (HDLC) or Multilink PPP (MLPPP) encapsulation on dialer interfaces.
- The dialer interface cannot be configured as a constituent link in a multilink bundle.
- The dialer interface can perform backup, dialer filter, and dialer watch functions, but these operations are mutually exclusive. You can configure a single dialer interface to operate in only one of the following ways:
    - As a backup interface—for one primary interface

- As a dialer filter
- As a dialer watch interface

The backup dialer interfaces are activated only when the primary interface fails. USB modem backup connectivity is supported on all interfaces except lsq-0/0/0.

The dial-on-demand routing backup method allows a USB modem connection to be activated only when network traffic configured as an “interesting packet” arrives on the network. Once the network traffic is sent, an inactivity timer is triggered and the connection is closed. You define an interesting packet using the dialer filter feature of the device. To configure dial-on-demand routing backup using a dialer filter, you first configure the dialer filter and then apply the filter to the dialer interface.

Dialer watch is a backup method that integrates backup dialing with routing capabilities and provides reliable connectivity without relying on a dialer filter to trigger outgoing USB modem connections. With dialer watch, the device monitors the existence of a specified route. If the route disappears, the dialer interface initiates the USB modem connection as a backup connection.

How the Device Initializes USB Modems

When you connect the USB modem to the USB port on the device, the device applies the modem AT commands configured in the `init-command-string` command to the initialization commands on the modem.

If you do not configure modem AT commands for the `init-command-string` command, the device applies the following default sequence of initialization commands to the modem: `AT S7=45 S0=0 V1 X4 &C1 E0 Q0 &Q8 %C0`. Table 57 on page 488 describes the commands. For more information about these commands, see the documentation for your modem.

Table 57: Default Modem Initialization Commands

Modem Command	Description
`AT`	Attention. Informs the modem that a command follows.
`S7=45`	Instructs the modem to wait 45 seconds for a telecommunications service provider (carrier) signal before terminating the call.
`S0=0`	Disables the auto answer feature, whereby the modem automatically answers calls.
`V1`	Displays result codes as words.

Table 57: Default Modem Initialization Commands *(Continued)*

Modem Command	Description
&C1	Disables reset of the modem when it loses the carrier signal.
E0	Disables the display on the local terminal of commands issued to the modem from the local terminal.
Q0	Enables the display of result codes.
&Q8	Enables Microcom Networking Protocol (MNP) error control mode.
%C0	Disables data compression.

When the device applies the modem AT commands in the `init-command-string` command or the default sequence of initialization commands to the modem, it compares them to the initialization commands already configured on the modem and makes the following changes:

- If the commands are the same, the device overrides existing modem values that do not match. For example, if the initialization commands on the modem include `S0=0` and the device's `init-command-string` command includes `S0=2`, the device applies `S0=2`.
- If the initialization commands on the modem do not include a command in the device's `init-command-string` command, the device adds it. For example, if the `init-command-string` command includes the command `L2`, but the modem commands do not include it, the device adds `L2` to the initialization commands configured on the modem.

NOTE: On SRX210 devices, the USB modem interface can handle bidirectional traffic of up to 19 Kbps. On oversubscription of this amount (that is, bidirectional traffic of 20 Kbps or above), keepalives do not get exchanged, and the interface goes down. (Platform support depends on the Junos OS release in your installation.)

USB Modem Configuration Overview

NOTE: USB modems are no longer supported for dial backup on SRX300, SRX320, SRX340, and SRX345 devices.

Before you begin:

1. Install device hardware. For more information, see the Getting Started Guide for your device.
2. Establish basic connectivity. For more information, see the Getting Started Guide for your device.
3. Order a US Robotics USB 56k V.92 Modem, model number USR Model 5637 (http://www.usr.com/).
4. Order a public switched telephone network (PSTN) line from your telecommunications service provider. Contact your service provider for more information.
5. Connect the USB modem to the device's USB port.

NOTE: When you connect the USB modem to the USB port on the device, the USB modem is initialized with the modem initialization string configured for the USB modem interface on the device.

a. Plug the modem into the USB port.

b. Connect the modem to your telephone network.

i.

Suppose you have a branch office router and a head office router each with a USB modem interface and a dialer interface. This example shows you how to establish a backup connection between the branch office and head office routers. See Table 58 on page 490 for a summarized description of the procedure.

Table 58: Configuring Branch Office and Head Office Routers for USB Modem Backup Connectivity

Router Location	Configuration Requirement	Procedure
Branch Office	Configure the logical dialer interface on the branch office router for USB modem dial backup.	To configure the logical dialer interface, see *Example: Configuring a USB Modem Interface*.

Table 58: Configuring Branch Office and Head Office Routers for USB Modem Backup Connectivity *(Continued)*

Router Location	Configuration Requirement	Procedure
	Configure the dialer interface `dl0` on the branch office router using one of the following backup methods: • Configure the dialer interface `dl0` as the backup interface on the branch office router's primary T1 interface `t1-1/0/0`. • Configure a dialer filter on the branch office router's dialer interface. • Configure a dialer watch on the branch office router's dialer interface.	Configure the dialer interface using one of the following backup methods: • To configure `dl0` as a backup for `t1-1/0/0` see "Example: Configuring Dialer Interfaces and Backup Methods for USB Modem Dial Backup" on page 496. • To configure a dialer filter on `dl0`, see "Example: Configuring Dialer Interfaces and Backup Methods for USB Modem Dial Backup" on page 496. • To configure a dialer watch on `dl0`, see "Example: Configuring Dialer Interfaces and Backup Methods for USB Modem Dial Backup" on page 496.
Head Office	Configure dial-in on the dialer interface `dl0` on the head office router.	To configure dial-in on the head office router, see *Example: Configuring a Dialer Interface for USB Modem Dial-In*.

If the dialer interface is configured to accept only calls from a specific caller ID, the device matches the incoming call's caller ID against the caller IDs configured on its dialer interfaces. If an exact match is not found and the incoming call's caller ID has more digits than the configured caller IDs, the device performs a right-to-left match of the incoming call's caller ID with the configured caller IDs and accepts the incoming call if a match is found. For example, if the incoming call's caller ID is 4085321091 and the caller ID configured on a dialer interface is 5321091, the incoming call is accepted. Each dialer interface accepts calls from only callers whose caller IDs are configured on it.

See Table 59 on page 492 for a list of available incoming map options.

Table 59: Incoming Map Options

Option	Description
accept-all	Dialer interface accepts all incoming calls. You can configure the `accept-all` option for only one of the dialer interfaces associated with a USB modem physical interface. The dialer interface with the `accept-all` option configured is used only if the incoming call's caller ID does not match the caller IDs configured on other dialer interfaces.
caller	Dialer interface accepts calls from a specific caller ID. You can configure a maximum of 15 caller IDs per dialer interface. The same caller ID must not be configured on different dialer interfaces. However, you can configure caller IDs with more or fewer digits on different dialer interfaces. For example, you can configure the caller IDs 14085551515, 4085551515, and 5551515 on different dialer interfaces.

You configure dialer interfaces to support PAP. PAP allows a simple method for a peer to establish its identity using a two-way handshake during initial link establishment. After the link is established, an ID and password pair are repeatedly sent by the peer to the authenticator until authentication is acknowledged or the connection is terminated.

Example: Configuring a USB Modem Interface

IN THIS SECTION

This example shows how to configure a USB modem interface for dial backup.

NOTE: USB modems are no longer supported for dial backup on SRX300, SRX320, SRX340, and SRX345 devices.

Requirements

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

Overview

In this example, you create an interface called as umd0 for USB modem connectivity and set the dialer pool priority to 25. You also configure a modem initialization string to autoanswer after a specified number of rings. The default modem initialization string is `AT S7=45 S0=0 V1 X4 &C1 E0 Q0 &Q8 %C0`. The modem command `S0=0` disables the modem from autoanswering the calls. Finally, you set the modem to act as a dial-in WAN backup interface.

Configuration

IN THIS SECTION

Procedure

CLI Quick Configuration

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

```
set interfaces umd0 dialer-options pool usb-modem-dialer-pool priority 25
set modem-options init-command-string "ATS0=2 \n" dialin routable
```

Step-by-Step Procedure

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

To configure a USB modem interface for dial backup:

1. Create an interface.

```
[edit]
user@host# edit interfaces umd0
```

2. Set the dialer options and priority.

```
[edit interfaces umd0]
user@host# set dialer-options pool usb-modem-dialer-pool priority 25
```

3. Specify the modem options.

```
[edit interfaces umd0]
user@host# set modem-options init-command-string "ATS0=2 \n"
```

4. Set the modem to act as a dial-in WAN backup interface.

```
[edit interfaces umd0]
user@host# set modem-options dialin routable
```

Results

From configuration mode, confirm your configuration by entering the `show interface umd0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interface umd0
modem-options {
init-command-string "ATS0=2 \n";
    dialin routable;
}
dialer-options {
    pool usb-modem-dialer-pool priority 25;
}
```

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

Verification

IN THIS SECTION

- Verifying the Configuration | 495

Confirm that the configuration is working properly.

Verifying the Configuration

Purpose

Verify a USB modem interface for dial backup.

Action

From configuration mode, enter the `show interfaces umd0 extensive` command. The output shows a summary of interface information and displays the modem status.

```
Physical interface:   umd0, Enabled, Physical link is Up
Interface index:      64, SNMP ifIndex: 33, Generation: 1
  Type: Async-Serial, Link-level type: PPP-Subordinate, MTU: 1504,
Clocking: Unspecified, Speed: MODEM
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps Internal: 0x4000
  Link flags     : None
  Hold-times     : Up 0 ms, Down 0 ms
  Last flapped   : Never
  Statistics last cleared: Never
 Traffic statistics:
   Input  bytes  :               21672
   Output bytes  :               22558
   Input  packets:                1782
   Output packets:                1832
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards: 0,
```

```
Resource errors: 0
  Output errors:
    Carrier transitions: 63, Errors: 0, Drops: 0, MTU errors: 0, Resource errors: 0
  MODEM status:
    Modem type                    : LT V.92 1.0 MT5634ZBA-USB-V92 Data/Fax Modem
(Dual Config) Version 2.27m
    Initialization command string : ATS0=2
    Initialization status         : Ok
    Call status                   : Connected to 4085551515
    Call duration                 : 13429 seconds
    Call direction                : Dialin
    Baud rate                     : 33600 bps
    Most recent error code        : NO CARRIER

  Logical interface umd0.0 (Index 2) (SNMP ifIndex 34) (Generation 1)
    Flags: Point-To-Point SNMP-Traps Encapsulation: PPP-Subordinate
```

Example: Configuring Dialer Interfaces and Backup Methods for USB Modem Dial Backup

IN THIS SECTION

This example shows how to configure a dialer interfaces and backup methods for USB modem dial backup.

NOTE: USB modems are no longer supported for dial backup on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Requirements

Before you begin, configure a USB modem for the device. See *Example: Configuring a USB Modem Interface*.

Overview

IN THIS SECTION

In this example, you configure a logical dialer interface on the branch office router for the USB modem dial backup. You then configure dial backup to allow one or more dialer interfaces to be configured as the backup link for the primary serial interface. To configure dialer watch, you first add a dialer watch interface and then configure the USB modem interface to participate as a dialer watch interface. The USB modem interface must have the same pool identifier to participate in dialer watch. Dialer pool name dw-pool is used when configuring the USB modem interface.

Topology

Configuration

IN THIS SECTION

Configuring a Dialer Interface for USB Modem Dial Backup

CLI Quick Configuration

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

```
set interfaces dl0 description USB-modem-backup encapsulation ppp
set interfaces dl0 unit 0 dialer-options activation-delay 60 deactivation-delay 30 idle-timeout
30 initial-route-check 30 pool usb-modem-dialer-pool
set interfaces dl0 unit 0 dialer-options dial-string 5551212
set interfaces dl0 unit 0 family inet address 172.20.10.2 destination 172.20.10.1
```

Step-by-Step Procedure

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

To configure a logical dialer interface on the branch office router for the USB modem dial backup:

1. Create an interface.

```
[edit]
user@host# edit interfaces dl0
```

2. Specify a description.

```
[edit interfaces dl0]
user@host# set description USB-modem-backup
```

3. Configure PPP encapsulation.

```
[edit interfaces dl0]
user@host# set encapsulation ppp
```

NOTE: You cannot configure Cisco High-Level Data Link Control (HDLC) or Multilink PPP (MLPPP) encapsulation on dialer interfaces used in USB modem connections.

4. Create the logical unit.

```
[edit interfaces dl0]
user@host# set unit 0
```

NOTE: You can set the logical unit to 0 only.

5. Configure the dialer options.

```
[edit interfaces dl0]
user@host# edit unit 0 dialer-options
user@host# set activation-delay 60
user@host# set deactivation-delay 30
user@host# set idle-timeout 30 initial-route-check 30 pool usb-modem-dialer-pool
```

6. Configure the telephone number of the remote destination.

```
[edit interfaces dl0 unit 0 dialer-options]
user@host# set dial-string 5551212
```

7. Configure source and destination IP addresses.

```
[edit]
user@host# edit interfaces dl0 unit 0
user@host# set family inet address 172.20.10.2 destination 172.20.10.1
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces dl0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interfaces dl0
    description USB-modem-backup;
encapsulation ppp;
unit 0 {
family inet {
    address 172.20.10.2/32 {
        destination 172.20.10.1;
        }
    }
        dialer-options {
            pool usb-modem-dialer-pool;
        dial-string 5551212;
            idle-timeout 30;
                activation-delay 60;
                deactivation-delay 30;
                initial-route-check 30;
                }
            }
```

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

Configuring a Dial Backup for a USB Modem Connection

CLI Quick Configuration

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

```
set interfaces t1-1/0/0 unit 0 backup-options interface dl0.0
```

Step-by-Step Procedure

To configure a dial backup for a USB modem connection:

1. Select the physical interface.

```
[edit]
user@host# edit interfaces t1-1/0/0 unit 0
```

2. Configure the backup dialer interface.

```
[edit]
user@host# set backup-options interface dl0.0
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces t1-1/0/0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host#  show interfaces t1-1/0/0
    encapsulation ppp;
    unit 0 {
        backup-options {
    interface dl0.0;
    }
}
```

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

Configuring a Dialer Filter for USB Modem Dial Backup

CLI Quick Configuration

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

```
set firewall family inet dialer-filter interesting-traffic term term1 from source-address
20.20.90.4/32
set firewall family inet dialer-filter interesting-traffic term term1 from destination-address
200.200.201.1/32
set firewall family inet dialer-filter interesting-traffic term term1 then note
set interfaces dl0 unit 0 family inet filter dialer interesting-traffic
```

Step-by-Step Procedure

To configure a dialer filter for USB modem dial backup:

1. Create an interface.

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

2. Configure the dialer filter name.

```
[edit]
user@host# edit family inet
user@host# edit dialer-filter interesting-traffic
```

3. Configure the dialer filter rule name and term behavior.

```
[edit]
user@host# edit term term1
user@host# set from source-address 20.20.90.4/32
user@host# set from destination-address 200.200.201.1/32
```

4. Configure the then part of the dialer filter.

```
[edit]
user@host# set then note
```

5. Select the dialer interface to apply the filter.

```
[edit]
user@host# edit interfaces dl0 unit 0
```

6. Apply the dialer filter to the dialer interface.

```
[edit]
user@host# edit family inet filter
user@host# set dialer interesting-traffic
```

Results

From configuration mode, confirm your configuration by entering the `show firewall family inet dialer-filter interesting-traffic` and `show interfaces dl0`commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show firewall family inet dialer-filter interesting-traffic
    term term1 {
from {
    source-address {
        20.20.90.4/32;
    }
        destination-address {
            200.200.201.1/32;
        }
        }
            then note;
        }
[edit]
    user@host# show interfaces dl0
unit 0 {
```

```
family inet {
    filter {
dialer interesting-traffic;
    }
        }
    }
```

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

Configuring a Dialer Watch for USB Modem Dial Backup

CLI Quick Configuration

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

```
set interfaces dl0 description dialer-watch unit 0 dialer-options watch-list 200.200.201.1/32
set interfaces dl0 unit 0 dialer-options pool dw-pool
set interfaces umd0 dialer-options pool dw-pool
```

Step-by-Step Procedure

To configure a dialer watch for USB modem dial backup:

1. Create an interface.

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

2. Specify a description.

```
[edit]
user@host# edit dl0
user@host# set description dialer-watch
```

3. Configure the route to the head office router for dialer watch.

```
[edit]
user@host# edit unit 0 dialer-options
user@host# set watch-list 200.200.201.1/32
```

4. Configure the name of the dialer pool.

```
[edit]
user@host# set pool dw-pool
```

5. Select the USB modem physical interface.

```
[edit]
user@host# edit interfaces umd0 dialer-options pool dw-pool
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces dl0` and `show interfaces umd0` commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show interfaces dl0
    dialer-options {
        pool dw-pool;
    }
[edit]
user@host# show interfaces umd0
    description dialer-watch;
        unit 0 {
    dialer-options {
        pool dw-pool;
            watch-list {
        200.200.201.1/32;
    }
        }
    }
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Configuration

Purpose

Verify the configuration output.

Action

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

Example: Configuring a Dialer Interface for USB Modem Dial-In

IN THIS SECTION

This example shows how to configure a dialer interface for USB modem dial-in.

NOTE: USB modems are no longer supported for dial-in to a dialer interface on SRX300, SRX320, SRX340, and SRX345 devices.

Requirements

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

Overview

To enable connections to the USB modem from a remote location, you must configure the dialer interfaces set up for USB modem use to accept incoming calls. You can configure a dialer interface to accept all incoming calls or accept only calls from one or more caller IDs.

If the dialer interface is configured to accept only calls from a specific caller ID, the system matches the incoming call's caller ID against the caller IDs configured on its dialer interfaces. If an exact match is not found and the incoming call's caller ID has more digits than the configured caller IDs, the system performs a right-to-left match of the incoming call's caller ID with the configured caller IDs and accepts the incoming call if a match is found. For example, if the incoming call's caller ID is 4085550115 and the caller ID configured on a dialer interface is 5550115, the incoming call is accepted. Each dialer interface accepts calls from only callers whose caller IDs are configured on it.

You can configure the following incoming map options for the dialer interface:

- `accept-all`—Dialer interface accepts all incoming calls.

  You can configure the `accept-all` option for only one of the dialer interfaces associated with a USB modem physical interface. The device uses the dialer interface with the `accept-all` option configured only if the incoming call's caller ID does not match the caller IDs configured on other dialer interfaces.

- `caller`—Dialer interface accepts calls from a specific caller ID— for example, `4085550115`. You can configure a maximum of 15 caller IDs per dialer interface.

  The same caller ID must not be configured on different dialer interfaces. However, you can configure caller IDs with more or fewer digits on different dialer interfaces. For example, you can configure the caller IDs 14085550115, 4085550115, and 5550115 on different dialer interfaces.

In this example, you configure the incoming map option as caller 4085550115 for dialer interface dl0.

Configuration

IN THIS SECTION

CLI Quick Configuration

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

```
set interfaces dl0 unit 0 dialer-options incoming-map caller 4085550115
```

Procedure

Step-by-Step Procedure

To configure a dialer interface for USB modem dial-in:

1. Select a dialer interface.

```
[edit]
user@host# edit interfaces dl0
```

2. Configure the incoming map options.

```
[edit]
user@host# edit unit 0 dialer-options incoming-map caller 4085551515
```

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

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

Verification

To verify the configuration is working properly, enter the `show interface dl0` command.

Example: Configuring PAP on Dialer Interfaces

IN THIS SECTION

This example shows how to configure PAP on dialer interfaces.

NOTE: Configuring PAP on dialer interfaces is no longer supported on SRX300, SRX320, SRX340, SRX345, SRX380, and SRX550HM devices.

Requirements

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

Overview

In this example, you specify a PAP access profile with a client username and a PAP password and select a dialer interface. Finally, you configure PAP on the dialer interface and specify the local name and password.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To configure PAP on the dialer interface:

1. Specify a PAP access profile.

```
[edit]
user@host# set access profile pap-access-profile client pap-access-user pap-password my-pap
```

2. Select a dialer interface.

```
[edit]
user@host# edit interfaces dl0 unit 0
```

3. Configure PAP on the dialer interface.

```
[edit]
user@host# set ppp-options pap local-name pap-access-user local-password my-pap
```

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

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

Verification

To verify the configuration is working properly, enter the `show interface dl0` command.

Example: Configuring CHAP on Dialer Interfaces

IN THIS SECTION

This example shows how to configure CHAP on dialer interfaces for authentication.

Requirements

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

Overview

In this example, you configure dialer interfaces to support CHAP for authentication. CHAP is a server-driven, three-step authentication method that depends on a shared secret password residing on both the server and the client. You specify a CHAP access profile with a client username and a password. You then specify a dialer interface as dl0. Finally, you enable CHAP on a dialer interface and specify a unique profile name containing a client list and access parameters.

Configuration

IN THIS SECTION

Procedure

Step-by-Step Procedure

To configure CHAP on a dialer interface:

1. Specify a CHAP access profile.

```
[edit]
user@host# set access profile usb-modem-access-profile client usb-modem-user chap-secret my-secret
```

2. Select a dialer interface.

```
[edit]
user@host# edit interfaces dl0 unit 0
```

3. Enable CHAP on the dialer interface.

```
[edit]
user@host# set ppp-options chap access-profile usb-modem-access-profile
```

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

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

Verification

To verify the configuration is working properly, enter the `show interface dl0` command.

Configuring DOCSIS Mini-PIM Interfaces

IN THIS SECTION

Data over Cable Service Interface Specifications (DOCSIS) define the communications and operation support interface requirements for a data-over-cable system. The topics below discuss the overview of DOCSIS Mini-PIM interface, its configuration details, and software features supported on DOCSIS Mini-PIM interfaces.

DOCSIS Mini-PIM Interface Overview

Data over Cable Service Interface Specifications (DOCSIS) define the communications and operation support interface requirements for a data-over-cable system. Cable operators use DOCSIS to provide Internet access over their existing cable infrastructure for both residential and business customers. DOCSIS 3.0 is the latest interface standard, allowing channel bonding to deliver speeds higher than 100 Mbps throughput in either direction, far surpassing other WAN technologies such as T1/E1, ADSL2+, ISDN, and DS3.

DOCSIS network architecture includes a cable modem with a DOCSIS Mini-*Physical Interface Module* (Mini-PIM) located at customer premises and a cable modem termination system (CMTS) located at the head-end or data center locations. Standards-based DOCSIS 3.0 Mini-PIM is interoperable with CMTS equipment. The DOCSIS Mini-PIM provides backward compatibility with CMTS equipment based on the following standards:

- DOCSIS 2.0
- DOCSIS 1.1
- DOCSIS 1.0

The cable modem interface of Mini-PIM is managed and monitored by CMTS through SNMP. This DOCSIS 3.0 Mini-PIM can be deployed in any multiple service operator (MSO) networks. The primary application is for distributed enterprise offices to connect to a CMTS network through the DOCSIS 3.0 (backward compatible to 2.0, 1.1, and 1.0) interface. The DOCSIS Mini-PIM uses PIM infrastructure developed for third-party PIMs.

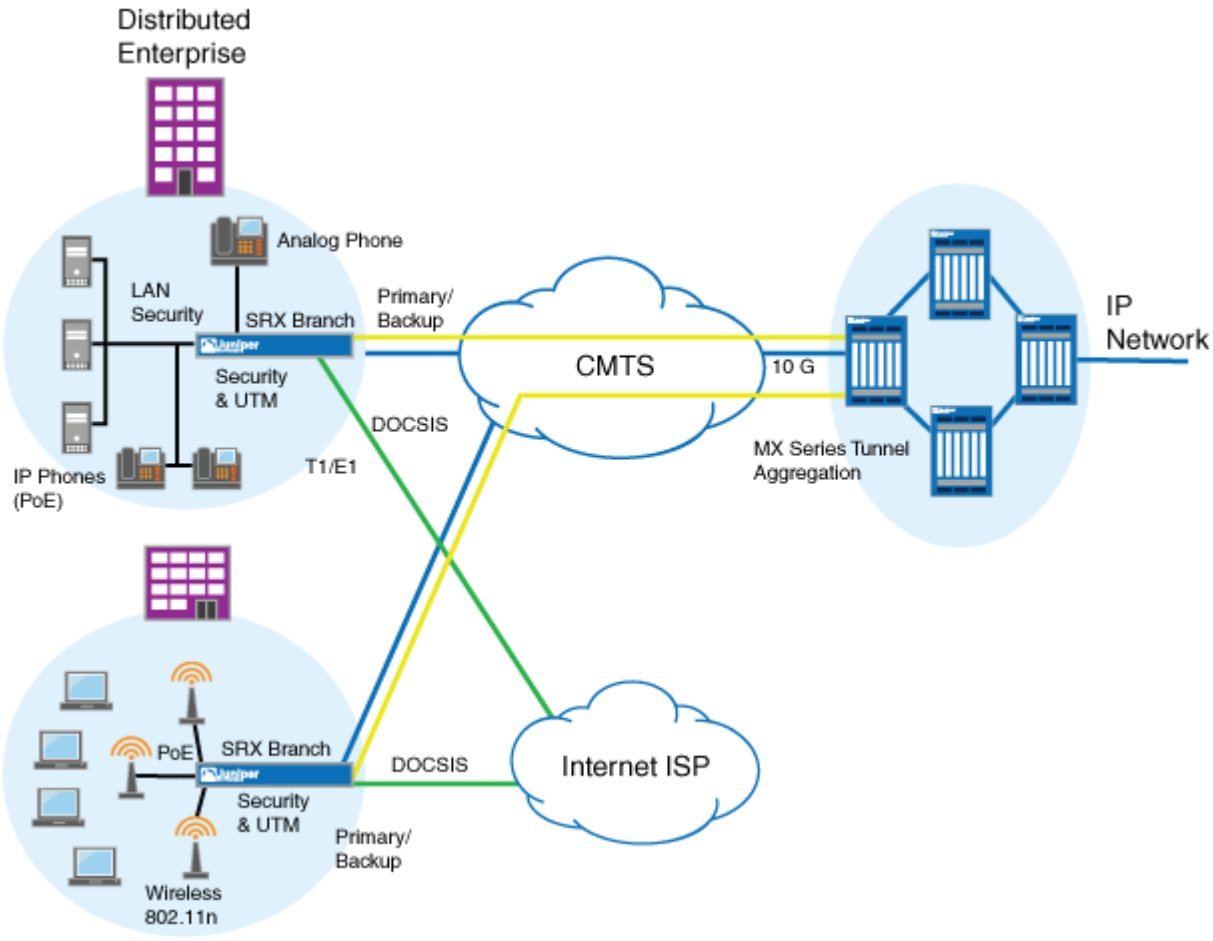
The Mini-PIM can also be used with encapsulations other than GRE, PPPoE, and IP-in-IP.

NOTE: The following interface trace options are supported:

- **all**—Enable all interface trace flags
- **event**—Trace interface events
- **ipc**—Trace interface IPC messages
- **media**—Trace interface media changes

CMTS manages and monitors the cable modem interface of then Mini-PIM through SNMP. This DOCSIS 3.0 Mini-PIM can be deployed in any multiple MSO network. Figure 25 on page 514 shows a typical use for this Mini-PIM in an MSO network.

Figure 25: Typical DOCSIS End-to-End Connectivity Diagram



Software Features Supported on DOCSIS Mini-PIMs

Table 60 on page 515 lists the software features supported on DOCSIS Mini-PIMs.

Table 60: Software Features Supported on DOCSIS Mini-PIMs

Software Feature	Description
DHCP and DHCPv6 clients	The DHCP and DHCPv6 clients are used to get the IP address from the CMTS using the DHCP protocol. DHCP is supported on IPv4 and IPv6. One of the main components of the configuration file is the static public IP address, which CMTS assigns to the cable modem. The management IP address is configured on the Mini-PIM's hybrid fiber coaxial (HFC) interface, which performs the following tasks: • Allows CMTS to execute remote monitoring and management of the Mini-PIM's cable interface. • Downloads the configuration file from CMTS and uses it for configuring the cable interface.
QoS support	The device is configured through the existing QoS CLI. Because the configuration on the Routing Engine and Mini-PIM is done together, the QoS configuration has to be consistent between the Routing Engine and the cable modem interface. The QoS mechanisms on the Routing Engine are decoupled from the QoS mechanisms on the Mini-PIM. The configuration file downloaded from CMTS contains parameters for primary and secondary flows. These parameters are programmed in the DOCSIS Mini-PIM. The Mini-PIM sends these parameters to the Routing Engine through the PIM infrastructure. The secondary flows are prioritized over primary flows in the DOCSIS Mini-PIM.
SNMP support	CMTS issues the SNMP requests that go to the cable modem. The DOCSIS MIB on the Routing Engine displays the Ethernet interface of the cable modem. The following features are supported on the DOCSIS Mini-PIM: • NAT support • Dying gasp support • Back pressure information
MAC address	The MAC address of the DOCSIS Mini-PIM is statically set at the factory and cannot be changed. The MAC address is retrieved from the Mini-PIM and assigned to the cable modem interface in Junos OS.

Table 60: Software Features Supported on DOCSIS Mini-PIMs *(Continued)*

Software Feature	Description
Transparent bridging	The DOCSIS Mini-PIM performs transparent bridging by sending the packets received on the Ethernet interface to the HFC interface and vice versa, without any modifications to the packet. All the other services such as webserver, DHCP server, and DNS server are disabled on the DOCSIS Mini-PIM during transparent bridging.

Example: Configuring the DOCSIS Mini-PIM Interfaces

IN THIS SECTION

This example shows how to configure DOCSIS Mini-PIM network interfaces.

Requirements

Before you begin:

- Establish basic connectivity. See the Quick Start for your device.
- Configure network interfaces as necessary. See "Example: Creating an Ethernet Interface" on page 130.

Overview

In this example, you configure the DOCSIS Mini-PIM interface as cm-2/0/0. You specify the physical properties by setting the interface trace options and the flag option. You then set the logical interface to unit 0 and specify the family protocol type as inet. Finally, you configure the DHCP client.

Configuration

IN THIS SECTION

- Procedure | 517

Procedure

CLI Quick Configuration

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

```
set interfaces cm-2/0/0 traceoptions flag all
set interfaces cm-2/0/0 unit 0 family inet dhcp
```

Step-by-Step Procedure

To configure the DOCSIS Mini-PIM network interfaces:

1. Configure the interface.

```
[edit]
user@host# edit interfaces cm-2/0/0
```

2. Set the interface trace options.

```
[edit]
user@host# set interfaces cm-2/0/0 traceoptions
```

3. Specify the flag option.

```
[edit]
user@host# set interfaces cm-2/0/0 traceoptions flag all
```

4. Set the logical interface.

```
[edit]
user@host# set interfaces cm-2/0/0 unit 0
```

5. Specify the family protocol type.

```
[edit]
user@host# set interfaces cm-2/0/0 unit 0 family inet
```

6. Configure the DHCP client.

```
[edit]
user@host# set interfaces cm-2/0/0 unit 0 family inet dhcp
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces cm-2/0/0` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show interfaces cm-2/0/0
    traceoptions {
    flag all;
    }
    unit 0 {
        family inet {
    dhcp;
    }
}
```

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

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the DOCSIS Interface Properties

Purpose

Verify that the DOCSIS interface properties are configured properly.

Action

From operational mode, enter the `show interfaces cm-2/0/0` command.

```
user@host> show interfaces cm-2/0/0 extensive
Physical interface: cm-2/0/0, Enabled, Physical link is Up
  Interface index: 154, SNMP ifIndex: 522, Generation: 157
  Link-level type: Ethernet, MTU: 1518, Speed: 40mbps
  Link flags     : None
  Hold-times     : Up 0 ms, Down 0 ms
   State   : OPERATIONAL, Mode: 2.0, Upstream speed: 5120000 0 0 0
   Downstream scanning: CM_MEDIA_STATE_DONE, Ranging: CM_MEDIA_STATE_DONE
   Signal to noise ratio: 31.762909 21.390018  7.517472 14.924058
   Power: -15.756125 -31.840363 -31.840363 -31.840363
   Downstream buffers used           : 0
   Downstream buffers free           : 0
   Upstream buffers free             : 0
   Upstream buffers used             : 0
   Request opportunity burst         : 0 MSlots
  Physical burst                    : 0 MSlots
   Tuner frequency                   : 555 0 0 0 MHz
   Standard short grant              : 0 Slots
   Standard long grant               : 0 Slots
   Baseline privacy state: authorized, Encryption algorithm: ????, Key length: 0
```

```
MAC statistics:                      Receive         Transmit
  Total octets                          1935             2036
  Total packets                            8                8
  CRC/Align errors                         0                0
  Oversized frames                         0
CoS queues     : 8 supported, 8 maximum usable queues
Current address: 00:24:dc:0d:76:19, Hardware address: 00:24:dc:0d:76:19
Last flapped   : 2009-11-10 19:55:40 UTC (00:16:29 ago)
Statistics last cleared: Never
Traffic statistics:
 Input  bytes  :                 710                   0 bps
 Output bytes  :                 866                   0 bps
 Input  packets:                   2                   0 pps
 Output packets:                   4                   0 pps
Packet Forwarding Engine configuration:
  Destination slot: 1
  Direction : Output
  CoS transmit queue               Bandwidth               Buffer Priority   Limit
                            %            bps     %           usec
  0 best-effort            95       38000000    95              0      low    none
  3 network-control         5        2000000     5              0      low    none
 Logical interface cm-2/0/0.0 (Index 69) (SNMP ifIndex 523) (Generation 134)
  Flags: Point-To-Point SNMP-Traps Encapsulation: ENET2
  Traffic statistics:
   Input  bytes  :                 710
   Output bytes  :                 806
   Input  packets:                   2
   Output packets:                   4
  Local statistics:
   Input  bytes  :                 710
   Output bytes  :                 806
   Input  packets:                   2
   Output packets:                   4
  Transit statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps
  Security: Zone: Null
  Flow Statistics :
  Flow Input statistics :
    Self packets :                    0
    ICMP packets :                    0
```

```
    VPN packets :                      0
    Multicast packets :                0
    Bytes permitted by policy :        0
    Connections established :          0
  Flow Output statistics:
    Multicast packets :                0
    Bytes permitted by policy :        0
  Flow error statistics (Packets dropped due to):
    Address spoofing:                  0
    Authentication failed:             0
    Incoming NAT errors:               0
    Invalid zone received packet:      0
    Multiple user authentications:     0
    Multiple incoming NAT:             0
    No parent for a gate:              0
    No one interested in self packets: 0
    No minor session:                  0
    No more sessions:                  0
    No NAT gate:                       0
    No route present:                  0
    No SA for incoming SPI:            0
    No tunnel found:                   0
    No session for a gate:             0
    No zone or NULL zone binding       0
    Policy denied:                     0
    Security association not active:   0
    TCP sequence number out of window: 0
    Syn-attack protection:             0
    User authentication errors:        0
  Protocol inet, MTU: 1504, Generation: 147, Route table: 0
    Flags: None
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 20.20.20/24, Local: 20.20.20.5, Broadcast: 20.20.20.255, Generation: 144
```

The output shows a summary of DOCSIS interface properties. Verify the following information:

- The physical interface is `Enabled`. If the interface is shown as `Disabled`, do either of the following:
  - In the CLI configuration editor, delete the `disable` statement at the `[edit interfaces` *`interface-name`*`]` level of the configuration hierarchy.
  - In the J-Web configuration editor, clear the `Disable` check box on the `Interfaces>`*`interface-name`* page.

- The physical link is `Up`. A link state of `Down` indicates a problem with the interface module, interface port, or physical connection (link-layer errors).
- The `Last Flapped` time is an expected value. The `Last Flapped` time indicates the last time the physical interface became unavailable and then available again. Unexpected flapping indicates likely link-layer errors.
- The traffic statistics reflect the expected input and output rates. Verify that the number of inbound and outbound bytes and packets matches the expected throughput for the physical interface. To clear the statistics and see only new changes, use the `clear interfaces statistics` *`interface-name`* command.

10

CHAPTER

Configuration Statements and Operational Commands

IN THIS CHAPTER

Junos CLI Reference Overview

We've consolidated all Junos CLI commands and configuration statements in one place. Read this guide to learn about the syntax and options that make up the statements and commands. Also understand the contexts in which you'll use these CLI elements in your network configurations and operations.

- Junos CLI Reference

Click the links to access Junos OS and Junos OS Evolved configuration statement and command summary topics.

- Configuration Statements
- Operational Commands